

3. Beschreibung der Systeme

Bei Beginn der Arbeit hatte sich der Verfasser für die Beschreibung der drei ausgewählten Programmpakete zum Ziel gesetzt, die Systeme nicht nur auf der Basis von Handbuchangaben zu beschreiben, sondern sie möglichst ausführlich in der Praxis und unter möglichst gleichen Bedingungen zu erproben. Dies Ziel konnte weitgehend erreicht werden, wo dies nicht möglich war, liegt die Verantwortung nicht beim Verfasser.

Zum einen verfügte der Verfasser zu Testzwecken nur in zwei Fällen über voll funktionsfähige Versionen der zu beschreibenden Systeme.⁴⁷ Im Falle von TINman war dies leider nicht möglich, da die damalige Vertriebsbeauftragte der Herstellerfirma sich als wenig kooperationsbereit erwies. Glücklicherweise kam es aber zu einer sehr intensiven Zusammenarbeit mit dem oben erwähnten OBI-Projekt „Einsatz eines Arbeitsplatzcomputers an Spezialbibliotheken mittlerer Größenordnung“, in dessen Rahmen TINman eingesetzt wird: das System konnte also an der Universitätsbibliothek Heidelberg intensiv erprobt werden, wenn auch auf einem anderen Rechner.

Zum anderen war geplant gewesen, die Testläufe mit realistischen Datenmengen durchzuführen: als grober Anhaltspunkt wurde der 'Durchschnittsbestand' einer Institutsbibliothek mit 10000 Titelaufnahmen zugrundegelegt.⁴⁸ Es war vorgesehen, eine mit dem System Bibliofile⁴⁹ zu erstellende Beispieldatei von 10000 Titeln im MARC-Format⁰ in die jeweiligen Systeme zu importieren und anschließend mit den so erzeugten Datenbeständen zu arbeiten. Dies hätte nach Maßgabe der

⁴⁷ Der Entwickler und Vertreiber von Allegro-C, Bernhard Eversberg, erwies sich als sehr kooperationsbereit; er stellte eine voll funktionsfähige Demo-Version und viele Hintergrundinformationen zur Verfügung. LIDOS ist von der Fachhochschule für Bibliotheks- und Dokumentationswesen in Köln erworben worden, so daß die Tests problemlos dort durchzuführen waren.

⁴⁸ Die Institutsbibliotheken an der Universität Bonn haben beispielsweise nach den im Bibliotheksführer gemachten Angaben einen Durchschnittsbestand von 10438 Bänden. Diese Zahl dient natürlich nur als Richtwert: die tatsächliche Streubreite reicht von 500 bis zu weit über 100000 Bänden.

⁴⁹ Bibliofile erlaubt den Zugriff auf die (als Compact Disk vorliegenden) MARC-Datensätze der Library of Congress.

so Mithin dem gängigsten internationalen Austauschformat.

Handbuch- und Produktinformationen in allen drei zu untersuchenden Fällen möglich sein müssen. Leider erwiesen sich dieses Vorhaben als nur zum Teil realisierbar: die Beispieldatei konnte wie geplant aufgebaut werden, und auch der Import der Datensätze war im Falle von Allegro-C problemlos möglich. Unter LIDOS war die Importfunktion zwar prinzipiell aktivierbar, konnte aber dennoch den Import der Beispieldatei nicht bewältigen. Bei TINman schließlich war die (als voll funktionsfähig beschriebene) Importfunktion nicht zu benutzen, wodurch alle Laufzeituntersuchungen unmöglich wurden.

Endlich konnten die Laufzeittests nicht wie vorgesehen auf zwei unterschiedlichen Rechnern durchgeführt werden: da der Aufbau der Testdateien eine äußerst langwierige und unsichere Prozedur darstellte, konnte diese Arbeit nur einmal vollständig gemacht werden. Die Funktionstests wurden aber auf beiden Rechnern durchgeführt.

Funktions- und Laufzeittests wurden auf dem zum IBM-PC/XT voll kompatiblen 'No-Name'-Rechner „NanoComp" durchgeführt. Laufzeitrelevante Spezifikationsdaten dieses Rechners sind:

- Prozessor V20 von NEC (schneller als der 8088 von Intel);
- Phoenix-BIOS;
- Taktfrequenz: 8 Mhz.;
- Mittlere Plattenzugriffszeit: 80 Millisekunden (sehr langsam!);
- Norton-Faktor: 3,2. Es wurde das Betriebssystem PC-DOS 3.20 verwendet.

Alle Funktions- und einige Laufzeittests wurden auf dem zum IBM-PC/AT kompatiblen Rechner „Acer" von Multitech durchgeführt. Laufzeitrelevante Spezifikationsdaten dieses Rechners sind:

- Prozessor Intel 80286;
- Award-BIOS;
- Taktfrequenz: 10 Mhz.;
- Mittlere Plattenzugriffszeit: 42.34 Millisekunden ;

- Norton -Faktor: 9,7. Es wurde das Betriebssystem MS-DOS 3.10 verwendet.

Die Funktionstests mit TINman wurden auf einem zum IBM-PC/AT kompatiblen 32-Bit Rechner, dem Compaq Deskpro 386, durchgeführt, da keine Laufzeiten ermittelt werden konnten, entfallen hier die entsprechenden Spezifikationsdaten.

Allegro-C

A Charakterisierung

Das System Allegro-C5¹ ist die einzige derzeit auf dem Markt befindliche genuin bibliothekarische deutsche Entwicklung. Es handelt sich um ein äußerst flexibles System zur Erfassung, Verwaltung und Auswertung bibliographischer und anderer Daten. Drei Eigenschaften können vorab als für Allegro charakteristisch festgehalten werden:

- bei der Entwicklung wurde großer Wert auf Portabilität, Transparenz und Integrierbarkeit von Allegro und auch von Teilfunktionen des Pakets als Module in komplexere Systeme gelegt; auch ist die Entwicklung der ohnehin schon sehr transparenten Software in seltenem Ausmaß von bibliothekarischer Seite durch den direkten Kontakt mit dem Entwickler, Bernhard Eversberg,² beeinflussbar ;
- die sehr weitgehenden Anpassungsmöglichkeiten von Allegro über komplexe Parametrierungen machen das System hochgradig flexibel, stellen jedoch an den Anwender erhebliche Anforderungen;
- die Anpassung an unterschiedliche bibliographische Standards und EDV-Erfassungsformate ist bei Allegro sehr weit fortgeschritten.

Das System Allegro ist tatsächlich eine Sammlung aufeinander abgestimmter, jedoch programmtechnisch völlig eigenständiger Module, die dem Anwender auf Ebene des Betriebssystems durch eine batch-Funktion (A.BAT) unter einer Oberfläche präsentiert werden. Jedes dieser Module ist separat ablauffähig und somit in eigene batch-Oberflächen oder auch in Anwendungsprogrammierungen integrierbar. Für die einzelnen Module existiert ein inzwischen weit entwickeltes System von Kommandozeilenparametern zur Ablaufsteuerung, die das Verhalten des Systems beim Aufruf von der Betriebssystemebene (bzw. aus batch-Oberflächen und auch Anwendungsprogrammierungen) bis ins Detail beeinflussen können; mit Hilfe dieser Parametersteuerung können Routineverarbeitungen zusammengefaßt, beispielsweise

¹ Im folgenden kurz als Allegro bezeichnet; der Zusatz „-C“ bezieht sich auf die Programmiersprache C, in der Allegro geschrieben ist.

² Universitätsbibliothek Braunschweig, Pockelstr. 13, 3300 Braunschweig.

umfangreiche Abfrageroutinen abgekürzt werden. Da die Betriebssystemebene erst beim Aufruf einzelner Module verlassen wird, können von der batch-Oberfläche auch DOS-Funktionen aufgerufen werden; aus eben diesem Grund kann vom Grundmenu aus ein weiteres vom Anwender frei definierbares Programm (etwa ein Texteditor) angesprochen werden, bei dessen Verlassen man sich im Grundmenu wiederfindet. Die entsprechende Änderung kann von dem mit MS-DOS Vertrauten leicht in der Datei A.BAT bewerkstelligt werden.

Allegro arbeitet mit einer nur auf den ersten Blick verwirrenden Vielzahl von Dateien: tatsächlich folgt deren Benennung einem klaren System und gibt über die Funktion jeder Datei schnellen Aufschluß. Grundsätzlich sind bei Allegro Dateien zur Steuerung des Programmablaufs (unter MS-DOS ablauffähige Dateien vom Typ *.BAT) und eigentliche Programmdateien (Typ *.EXE) einerseits von den Dateien zur eigentlichen Datenverwaltung andererseits zu unterscheiden. Prinzipiell kann Allegro mit einer Vielzahl unterschiedlicher Kategoriensysteme arbeiten. Hierbei arbeiten die Programmmodule jeweils mit einer Konfigurationsdatei (*.CFG) zusammen, die grundlegende Informationen über das verwendete Kategoriensystem enthält. Mitgeliefert werden die Dateien A.CFG (NMN-Schema), N.CFG (NZN-Schema), S.CFG (SWB-MAB), P.CFG ('PC-MAB', das 'Heidelberger Schema') und D.CFG (DB-MAB). Jeweils der erste Buchstabe einer solchen Konfigurationsdatei dient nun auch zur Kennzeichnung der von der dieser Konfiguration abhängigen Verarbeitungs- und Steuerdateien; (nichtindizierte) Allegro-Dateien sind vom Typ *.?LG, Dateien zur Indexgenerierung vom Typ *.?PI, die eigentlichen Indexdateien vom Typ *.?DX, indizierte Datenbestände haben die Kennung *.?LD, Dateien zur Importsteuerung sind vom Typ *.?IM, solche zur Exportsteuerung vom Typ *.?PR; endlich gibt es konfigurationsspezifische Phrasendateien vom Typ *.?PH und Stopwortlisten vom Typ *.?PT. In allen diesen Fällen steht das Fragezeichen für den Kennbuchstaben der Konfiguration: so ist etwa die Datei OCLC.DIM eine Steuerungsdatei zum Import von OCLC-Daten bezogen auf das Kategoriensystem DB-MAB, die Datei RSWK.SDX wäre eine Datei zur Indexgenerierung bezogen auf die Konfiguration SWB-MAB mit Auswertung der Beschlagwortung nach den RSWK in den Aufnahmen für die Indexgenerierung. Eine große Zahl solcher Steuerungsdateien wird mitgeliefert, weitere sind gegebenenfalls nach einer gewissen Einarbeitungszeit (s.u.) völlig frei erstellbar.

B Entwicklung, Vertrieb

Allegro wird seit Ende 1985 an der Universitätsbibliothek Braunschweig von Dipl.-Ing. Bernhard Eversberg entwickelt, seine Vorgeschichte reicht jedoch bis in das Jahr 1980 zurück, denn das System in seiner jetzigen Form geht in manchen Bereichen immer noch auf die Konzeption seines weiterhin an der Universitätsbibliothek Braunschweig im Einsatz befindlichen Vorgängers Allegro84 zurück.⁵³ Allegro84 lief auf CBM-Rechnern der Firma Commodore. Es ist symptomatisch für die schwere Einschätzbarkeit der Entwicklungen im Microrechnerbereich, daß Eversberg selbst noch in einem Aufsatz von 1985⁵⁴ davon ausging, mit dem Beginn des Umschreibens von Allegro84 durchaus noch drei Jahre warten zu können, um dann gegen 1990 eine auf neuer Hardware lauffähige Version zu haben, dann jedoch Ende 1985 doch mit der Umstellung begann, nachdem sich der Industriestandard der IBM-kompatiblen Microrechner wohl doch stärker als ursprünglich noch in Eversbergs Aufsatz von 1985 erwartet etabliert hatte.⁵⁵ Um jedoch bei der Neuentwicklung in keine Sackgasse zu geraten, hat Eversberg von vornherein auf Portabilität großen Wert gelegt. Die Entwicklung – eine Umsetzung des Konzeptes von Allegro84 auf die C/MS-DOS Ebene, allerdings mit teilweise grundlegenden Veränderungen und Verbesserungen – ging zügig voran⁵⁶ und kann zumindest in den Kernfunktionen als abgeschlossen gelten, obwohl im Detail gerade in den Monaten November 1988 bis Februar 1989 sehr viele Verbesserungen erfolgt sind. Im Vergleich zu früheren, bisweilen erheblich 'absturzgefährdeten' Versionen kann nun davon ausgegangen werden, daß Allegro (in der derzeitigen Version 9.3) weitestgehend fehlerfrei ist.

Implementiert ist Allegro derzeit bei ungefähr 30 deutschen Anwendern. Zu diesen zählen neben der Universitätsbibliothek Braunschweig auch Hochschulanwender wie etwa die Universitätsbibliothek Oldenburg, die Universität Mannheim und die Hochschule der Bundeswehr

⁵³ Die Entwicklungsgeschichte von Allegro84 und Allegro ist durch die unter C genannten Aufsätze von EVERSBERG und die Arbeit von BRANNEMANN (1985) gut dokumentiert.

⁵⁴ EVERSBERG 1985, S. 589.

⁵⁵ Dies deutet Eversberg selbst in seinem Beitrag von 1988 als Grund an; vgl. EVERSBERG (1988), S. 19.

⁵⁶ Wie in EVERSBERG (1986a-c) und EVERSBERG (1987) dokumentiert.

in München . Unter UNIX ist Allegro in der Herzog August Bibliothek Wolfenbüttel implementiert, Hauptanwendung ist dort die retrospektive Altbestandserfassung.⁵⁷ An der TIB in Hannover werden derzeit vor allem CD-Anwendungen mit Allegro erprobt (wobei die Importfunktionen des Programms stark zum Tragen kommen) . Im Auftrag der Universitätsbibliothek Heidelberg hat Eversberg eine Anpassung erstellt (die Konfiguration P.CFG, die die Vorgaben des aus dem DBI-Projekt stammenden Pflichtenheftentwurfes von MÜNNICH berücksichtigt), die inzwischen das Teststadium hinter sich hat und bei der sinologischen Institutbibliothek an der Universität Heidelberg zum Einsatz kommen wird.

Nachdem im vergangenen Jahr das alte, noch aus der Architektur von Allegro84 stammende Zugriffskonzept durch ein Binärbaumalgorithmus ersetzt wurde, kann jetzt als das Hauptvorhaben im Rahmen der geplanten Weiterentwicklung von Allegro die Umstellung der derzeitigen Eindateienstruktur auf eine auch von Eversberg als programmtechnisch besser anerkannte funktionale Aufteilung der Daten auf mehrere Dateien gelten. Dies könnte dann auch eine echte 'authority control' ermöglichen.⁵⁸

Die bisher weitgehend effiziente und umsichtige Entwicklung von Allegro stimmt für die Zukunft des Systems dennoch nur verhalten optimistisch. Zwar wird Allegro mit einiger Wahrscheinlichkeit im niedersächsischen Verbund als Erfassungs- und Manipulationssoftware eine Rolle spielen, leider aber hatte sich die von Eversberg noch im Juni 1987 gehegte Hoffnung, die Entwicklung des Systems könnte durch die Aufnahme in das DBI-Projekt zur Erprobung von Microcomputer - Software einen kräftigen Schub erhalten, zerschlagen.⁵⁹ Es kann nicht ausgeschlossen werden, daß die Entwicklung des Systems aus Mangel an personellen und finanziellen Ressourcen nicht schnell genug vorangetrieben werden kann, denn zur Zeit deutet sich zwar eine tragfähige Lösung für den Vertrieb des Pakets an (s.u.), doch liegt die Entwicklungsarbeit immer noch weitgehend allein in den Händen von Bernhard Eversberg; viel wird also davon abhängen, wieviele Anwender Allegro

findet, die gegebenenfalls auch die Entwicklungsarbeit mittragen oder zumindest fördern können.

Über den Vertrieb von Allegro besteht zur Zeit noch keine restlose Klarheit: beabsichtigt ist der Vertrieb durch die Firma DABIS,⁶⁰ doch steht die Zustimmung des zuständigen Landesministeriums in Hannover zu diesem Vertriebskonzept noch aus.

Literatur

Die Entwicklung und den Einsatz des Vorgängersystems, Allegro84, dokumentieren ausführlich die Beiträge von EVERSBERG (1980, 1981 a-d, 1982 a-f, 1983 a-c und 1984); eine ausführliche Beschreibung gibt BRANNEMANN (1985) auf S. 71-85, außerdem ist Allegro84 in der annotierten Marktübersicht bei EVERSBERG (1985) auf S. 593-594 berücksichtigt.

Die Entwicklungsstadien von Allegro selbst zeichnen Aufsätze von EVERSBERG (1986 a-c, 1987 a-c sowie 1988) nach.

Kurze Funktionsbeschreibungen von Allegro erschienen in der 'Programmbörse' von ABI-Technik (6. 1986, 2; S. 155) und in dem Aufsatz von STOCK (1988), diese beruhen in beiden Fällen auf Herstellerangaben.

BRANNEMANN (1987) hat Allegro in seine Untersuchung mit einbezogen, das System ist also in der dort auf S. 51 wiedergegebenen tabellarischen Übersicht berücksichtigt, allerdings in einer älteren, gerade in puncto Systemstabilität noch nicht ausgereiften Version, auch das neue Zugriffskonzept ist dort noch nicht berücksichtigt .

Eversberg hat sein System selbst mit zwei anderen verglichen . Ein (sehr kritischer) Vergleich mit LIDOS findet sich in EVERSBERG (1986c), Ergänzungen und mildernde Korrekturen in EVERSBERG (1987a). Einen Vergleich von Allegro und BIBLIOGRAMM enthält EVERSBERG (1986b).

Es existiert außerdem ein Handbuch zu Allegro (Allegro-C).

⁵⁷ Vgl. dazu RATH-BECKMANN, S. 349, außerdem RUPPELT, S. 91.

⁵⁸ Vgl. dazu EVERSBERG (1988), S. 23 sowie Punkt 2.1. dieses Abschnittes.

⁵⁹ Vgl. dazu EVERSBERG (1987d), S. 12; s. außerdem den Beitrag von SCHMIDT (1988) und die Ausführungen in dieser Arbeit.

⁶⁰ P11lrn11ille 71, 2000 Hamburg.

D Beschreibung

1 Hardwareseitige und allgemein EDV-technische Bedingungen

1.1. Allegro ist in der kompilierten Form auf allen zum IBM-PC/XT und AT kompatiblen Rechnern, auf dem Siemens PC-D sowie auf allen das Betriebssystem UNIX verwendenden Rechnern lauffähig. Ein AT-Rechner ist jedoch mit Blick auf die Laufzeiten gerade bei der Volltextsuche empfehlenswert. Eine Version für den Atari-ST ist in Arbeit, allerdings mit wesentlich geänderten Funktionen. Die Codierung in der Sprache C macht Allegro zusätzlich prinzipiell auf allen Rechnern mit C-Compiler implementierbar.

1.2. Allegro benötigt aufgrund seiner stark modularen Struktur nur die als Minimalstandard auf jedem PC ohnehin verfügbaren 256 KB Hauptspeicher. Bei der gegenwärtigen Architektur von Allegro führt ein größeres Speichervolumen immer dann zu einer erheblichen Beschleunigung der Dateizugriffe (je nach Plattenzugriffszeit bis zu 30%), wenn die dann eingerichtete RAM-Disk sämtliche Datenbestände und Indexdateien aufnehmen kann; anderenfall steht bei einem Speicherausbau auf 640 KB viel Platz für speicherresidente Programme bleibt. Im Test gab es auf einem XT-kompatiblen Rechner auch in 'Koexistenz' mit 330 KB(!) einnehmenden speicherresidenten Programmen keine Ablaufprobleme.

1.3. Allegro ist voll mehrplatz- und netzwerkfähig.

1.4. Allegro ist derzeit unter MS-DOS (ab Version 2.0) und UNIX verfügbar.

1.5. Allegro ist in C programmiert, die MS-DOS Version ist unter Turbo-C kompiliert und greift nur in ganz wenigen Fällen auf Routinen des Betriebssystems zurück. C ist derzeit eine der am weitesten verbreiteten Hochsprachen, die zudem aufgrund ihrer relativen Nähe zur Maschinensprache einen sehr schnellen Code erzeugt (allerdings daher auch für den Neuling größere Anpassungsschwierigkeiten mit sich bringt als etwa BASIC oder auch PASCAL). Durch die Programmierung in C ist Allegro äußerst portabel, wenn auch langsamer als in Assembler programmierte Systeme.

1.6. Die übersichtliche Architektur, das einleuchtende System der Dateibenennungen und die klare modulare Trennung der Funktionsbereiche machen Allegro für den halbwegs MS-DOS-Kundigen sehr transparent, die Programmierung in C macht das System für die Programmierung von Schnittstellen zu anderen Anwendungen relativ offen, so daß etwa die Einbindung in integrierte Systeme möglich ist.⁶¹

1.7. Eine Online-Schnittstelle für den Datenaustausch existiert nicht und ist auch in der Planung nicht prioritär.

2 Datenbanktechnische Spezifikationen

2.1. Grenzwerte:

2.1.1. Die Anzahl der Datensätze pro Datei wird nur durch die Kapazität des Massenspeichers begrenzt, kann also im Prinzip beliebig groß werden, wobei ein Datensatz einer Titelaufnahme gleichzusetzen ist. Da zudem beliebig viele Allegro-Dateien (Kennung *.?LG) unter einer gemeinsamen Indexdatei in einer logischen Einheit organisierbar sind, bietet sich gegebenenfalls eine funktionale Aufteilung etwa thematisch oder nach Bearbeitern unterschiedener Datensätze auf verschiedene Dateien an, so daß für unterschiedliche Vorhaben unterschiedliches Datenmaterial zusammengestellt werden kann.

2.1.2. Die Anzahl der Zeichen pro Datensatz unterliegt keiner Beschränkung.

2.1.3. Die absolute Feldzahl unterliegt ebenfalls keiner Beschränkung, da zwar maximal 100 Datenfelder (= bibliographisch-bibliothekarische Kategorien) möglich sind, diese aber beliebig oft wiederholt und automatisch durch Zusatzkennungen unterschieden werden können.

2.1.4. Ein Feld kann maximal 2 000 Zeichen enthalten.

⁶¹ Ein erster konkreter Beleg für diese bislang nur theoretisch sichtbaren Vorzüge der absoluten Offenheit von Allegro-C stellt die Tatsache dar, daß die Firma DABIS für ihr System BIS-LOK, das bislang über keine Erfassungs- und Import-Exportschnittstellen verfügt, den Einsatz der entsprechenden Funktionen von Allegro derzeit ernsthaft ins Auge gefaßt hat (mündliche Information von Herrn Eversberg und der Fa. DABIS).

2.1.5. Es kann nach beliebig vielen frei definierbaren Datenfeldern gleichzeitig geordnet werden.

2.2. Allegro verwendet eine eigene, speziell entwickelte Datenbankstruktur, die bis zu fünffach gestufte Aufnahmen ermöglicht. Physisch sind die Daten unter Allegro in einer einzigen Datei indexsequentiell organisiert, die im Modus der Volltextsuche ebenfalls sequentiell durchlaufen wird. Der Schnellzugriff geschieht über Indexdateien (*.?DX).

2.3. Feld- und Satzlängen sind völlig flexibel.

2.4. Es können bis zu 100 (beliebig duplizierbare) Felder völlig frei mit beliebigen Benennungen gebildet werden. Dies geschieht durch Änderungen in den Dateien vom Typ *.CFG und ist mit jedem Texteditor zu bewerkstelligen, der reinen ASCII-Code erzeugt; hierbei sind kaum EDV-spezifischen Vorkenntnisse erforderlich.

2.5. Feldlängen sind nicht determinierbar.

2.6. Allegro nutzt den Platz im Massenspeicher ökonomisch aus: es wird nur soviel Platz belegt, wie tatsächlich von den eingegebenen Daten benötigt wird. Der für die Indexdateien benötigte Platz hängt von der Komplexität der Steuerdatei *.?PI ab; die beim Test verwendete, recht komplexe Indexstruktur führte zu einem zusätzlichen Platzbedarf in Höhe von ca. 30% der Bezugsdatei.

3. Regelwerks- und Kategorienanforderungen

3.1. Eine Strukturierung der Datensätze nach dem von MÜNNICH entwickelten Format ist möglich.

3.2. Es sind bis zu fünffach gestufte hierarchische Aufnahmen mit beliebig vielen Bandauftragungen möglich.

3.3. Stücke einer gezählten Serie sind nicht über den Gesamttitel verknüpfbar, es sei denn, man wählt für gezählte Serien die hierarchische Aufnahmestruktur für mehrbändige Werke – dann sind aber die Bände nicht isoliert recherchierbar. Allerdings besteht die Möglichkeit, durch die Vergabe gesonderter Identifikationsnummernkreise beispielsweise für Gesamttitel die Voraussetzungen für die hierarchische Verknüpfung der Einzelbände etwa für die Weiterverarbeitung im Großrechner zu schaffen.

4. Dateneingabe, Importfunktion

4.1. Ergonomie/Akzeptanz:

4.1.1. Allegro bietet dem Benutzer eine funktionale und durchdachte Bearbeitungsoberfläche ohne überflüssige Elemente. Allerdings ist bisweilen die Darstellung der (oft dichtgedrängten) Informationen nicht sehr übersichtlich.⁶² Die routinemäßig bei der Dateneingabe vom System „abzufragenden“ Kategorien können auch hinsichtlich der Reihenfolge der Abfrage frei definiert werden, nicht besetzte Kategorien verschwinden wieder vom Bildschirm, so daß die Anzeige auch bei komplexen Aufnahmeformaten übersichtlich bleibt. Nicht automatisch abgefragte Kategorien können unter Voranstellung der Kategorienkennung in freier Reihenfolge eingegeben werden. Es kann zwischen einem automatischem Abfragemodus (in dem in frei bestimmbarer Reihenfolge ebenfalls frei wählbare Kategorien aus der Gesamtkonfiguration auf dem Bildschirm zur Eingabe vorgeschlagen werden; nicht besetzte Kategorien werden wieder ausgeblendet) und Befehlsmodus gewählt werden; eine rein maskenorientierte Bearbeitung gibt es bei Allegro nicht. Gleiches gilt für Korrekturen und Änderungen: sie können bildschirmorientiert im Kontext der Gesamtaufnahme geschehen oder unter Voranstellung der jeweiligen Kategorienkennung frei eingegeben werden.

4.1.2. Die Form der Bildschirmanzeige kann vom Anwender nicht gestaltet werden, wohl aber können die anzuzeigenden Elemente auch in ihrer Reihenfolge bestimmt werden, sie können zudem für die Anzeige frei benannt und mit Erläuterungstexten versehen werden. Dies geschieht über eine Änderung in den Dateien vom Typ *.CFG.

4.1.3. Es können satz- und feldbezogene Daten aus schon bearbeiteten Aufnahmen übernommen werden. Dies geschieht über einen einfach zu handhabenden Befehl.

4.1.4. An sonstigen besonderen Erfassungshilfen existieren:

⁶² Es ist allerdings vorgesehen, auch die Bildschirmgestaltung von Allegro für den Anwender über Parametrisierungsmöglichkeiten zu öffnen; mit einer Implementierung der entsprechenden Funktionen ist nach Auskunft von Eversberg; sehr bald 1.11 r<-rhncn.

- ein Phrasenspeicher, in dem häufig wiederkehrende Eingabeelemente zur Mehrfachverwendung abgelegt werden können;
- eine Funktion zum Suchen und Ersetzen beliebiger Zeichenketten in der aktuellen Aufnahme oder in ausgewählten Aufnahmegruppen.

4.2. Routinen zur Sicherung der Datenhomogenität:

4.2.1. Authority-control ist angesichts der Eindateienstruktur von Allegro derzeit nicht möglich: ein Datensatz entspricht hier einer Titelaufnahme.⁶³

4.2.2. Über eine Änderung in den Dateien vom Typ *.CFG ist es möglich, jedem Eingabefeld eine der folgenden Prüfroutinen zuzuordnen:

- ungeprüfte Annahme;
- bei Artikel am Anfang kann routinemäßig die Abfrage erfolgen, ob ein Nichtsortierzeichen gesetzt werden soll (im Titelfeld von Interesse);
- Prüfung, ob ' ' im Eingabetext vorkommt (bei Namenseintragungen);
- Prüfung, ob ' ; ' im Eingabetext vorkommt (bei Zählungen);
- Plausibilitätsprüfung der Jahreszahl (Zahlen zwischen 1450 und 1999);

⁶³ Es wäre zwar theoretisch möglich, beispielsweise getrennte Namensdateien zu führen und zugleich mit der Hauptdatei zu verwalten; da das Schnellzugriffsprogramm von Allegro mit mehreren Dateien gleichzeitig arbeiten kann, wäre ein Blättern in der Namensdatei während der Erfassungsarbeit in der Hauptdatei denkbar, doch ist derzeit weder eine automatische Überführung der gefundenen Namen in die Hauptdatei noch eine Verknüpfung über Identifikationsnummern möglich. Eine weitere provisorische Abhilfe stellt ein im Handbuch (1-7) empfohlenes Verfahren dar: die Möglichkeit, Registereinträge (und damit auch vorhandene Namensansetzungen) als 'Phrasenbausteine' zwischenspeichern, erlaubt die Übernahme dieser Namensformen in neue Aufnahmen. Eversberg sieht jedoch selbst die Änderung der Dateistruktur als notwendig an, prinzipiell ist bei der Neuprogrammierung wohl auch von vornherein an authority-control gedacht gewesen: „Intern sind die Daten und das Zugriffsprogramm aber bereits so gestaltet, daß z.B. Namen oder Schlagwörter auch durch Identifikationsnummern ersetzt und in getrennten Dateien unter diesen Nummern abgespeichert werden könnten“ (EVERSBERG 1988, S. 23).

- Berechnung der ISBN-Prüfziffer.

Die Definition weiterer Prüfroutinen und Plausibilitätskontrollen durch den Anwender ist nicht möglich.

4.2.3. Pflichtkategorien und fakultative Bereiche werden unterschieden. Über eine Änderung in den *.CFG-Dateien kann jedem Feld der Status einer Pflicht- oder einer Hinweiskategorie (bibliographische Kernkategorien, die jedoch nicht Pflichtkategorien sind) frei zugeordnet werden. Das System gibt bei der Eingabe eine differenzierte Warnungsmeldung auf dem Bildschirm aus, sobald eine der so gekennzeichneten Kategorien fehlt.

4.2.4. Regelwerkshilfen sind derzeit nicht realisiert. Sie sind aber durch entsprechende Änderungen in den Dateien, die Hilfstexte enthalten (Dateien mit dem Anfangsbuchstaben H und ohne Kennung, an das H kann sich beispielsweise eine Nummer schließen, die damit zugleich zum Code zum Aufruf der entsprechenden Hilfstexte wird) realisierbar. Auf diese Weise sind nicht nur globale Hinweise, sondern auch situationsorientierte Hilfen einfach mit einem Texteditor zu erstellen.

4.3. Datenfelder sind beliebig oft belegbar (theoretisch bis zu 100mal).

4.4. Es ist der Zeichensatz des jeweiligen Rechners voll verfügbar.

4.5. Auch bei großen Datenmengen verlängerte sich die zum Speichern eines neuen Datensatzes notwendige Zeit nicht meßbar, wie von der Eindateienstruktur von Allegro her auch zu erwarten war. Bei der Erfassung im Schnellzugriffsmodus liegt die Einarbeitungszeit für eine neue Aufnahme bei gleichzeitiger Indexgenerierung und -aktualisierung auch bei 10000 Sätzen noch um 5 Sekunden.

4.6. Importfunktion:

4.6.1. Die Importfunktion, im Grunde nur ein Spezialfall der hochentwickelten Möglichkeiten der Datenumformung unter Allegro, stellt zusammen mit den Exportmöglichkeiten einen der Glanzpunkte des Systems dar.⁶⁴ Allerdings ist – wie auch im Falle der Exportfunktionen –

⁶⁴ Der im Beitrag EVERSBERG 1987b deutlich anklingende Stolz des Entwicklers ist also durchaus berechtigt.

u bemerken, daß die hier möglichen Datenumformungen von Definitionen in einer eigens geschaffenen „Datenmanipulationssprache“⁶⁵ gesteuert werden, daß also konkret gesprochen elementare Programmierkenntnisse zumindest von Nutzen sind. Die betreffende Befehlssprache ist tatsächlich nicht 'schwer' erlernbar, vergleicht man sie mit Hochsprachen wie BASIC, PASCAL oder gar C. Wer jedoch über keinerlei Erfahrung in der Programmierung verfügt, dürfte eine gewisse Einarbeitungszeit benötigen. Der Anwender bewegt sich hier im Grunde in der Grauzone zwischen Softwareanpassung und Programmierung.

Sieht man aber einmal von diesen 'Hürden' der Konvertierungsdefinition ab, so ist mit der Import-Funktion von Allegro alles nur Denkbare machbar: tatsächlich handelt es sich hier um einen komplexen Prozeß, der Daten in zwei Stufen transformiert. Generell werden die Ausgangsdaten in einem ersten Schritt im Arbeitsspeicher in das Allegro-Internschema transformiert (und damit auf die gerade gültige Konfiguration abgebildet); in einem zweiten Schritt werden diese Daten in bis zu drei Zieldateien mit beliebig definierbaren Kategorienschemata geschrieben. Der eigentliche Importvorgang ist somit nur ein Sonderfall dieses Umformungsprozesses, wobei der zweite Schritt der Verarbeitung entfällt: die Daten werden dabei in der im Arbeitsspeicher erzeugten Form (der gerade gültigen Kategorienkonfiguration) in eine Zieldatei geschrieben – ebenso gut ist es aber möglich, z. B. MARC-Ausgangsdaten in einem Umformungslauf gleichzeitig in eine nach SWB-MAB strukturierte Datei zu schreiben (ein Importvorgang), auf dem Drucker auszugeben und schließlich auf derselben Datenbasis eine 1\xtdatei (etwa in Form einer Literaturliste) zur Weiterverarbeitung mit 111e111 Textsystem zu erstellen. Mit diesem Verfahren können im Prinzip 1,()liebi g strukturierte Daten in eine ebenfalls frei gestaltbare Zielstruktur überführt werden, wobei sogar sehr schwach strukturierte Ausgangsdateien bearbeitet werden können, sofern sie nur mindestens über eine konsequent gehandhabte Interpunktion oder regelmäßige Umbruchmerkmale als Indikatoren für Feld- und Satzgrenzen verfügen; selbst die Umformung von Daten, die nicht über wiederkehrende Feldkennungen, sondern über Steuerverzeichnisse am Beginn der Datei verwaltet

⁶⁵ So Tlvcrsbcrgs eigene Formulierung in EVERSBERG (1987b), S. 635.

werden, ist mit Allegro möglich.⁶⁶ Mit einem solchen Importvorgang können komplexe Zeichenaustauschprozeduren verbunden werden, die beispielsweise die Ersetzung nicht RAK-gerechter Interpunktionszeichen in Fremdformaten durch die von RAK vorgeschriebenen Zeichen, das Austauschen fremdsprachlicher Bezeichnungen gegen die deutschen Äquivalente und endlich das Eliminieren nicht benötigter Elemente vornehmen. Die Funktion wurde mit sehr komplex strukturierten Daten im Biblofile-MARC Format getestet und erwies sich als voll realisiert.⁶⁷

Da der Importvorgang eine Eigenheit von Allegro, die weitgehenden Anpassungsmöglichkeiten über Parametrierungen, recht gut illustriert, soll er an dieser Stelle am Beispiel der für die vorliegende Untersuchung durchgeführten Umformung von Biblofile-MARC-Daten in das Format des Südwestverbundes (SWB-MAB) etwas detaillierter angesprochen werden.⁶⁸ Die Konvertierungstabellen (Namensform unter MS-DOS: *.?IM, hier MARCBF.SIM) werden mit Hilfe obenerwähnter Befehlssprache erstellt und haben folgenden klaren Aufbau:

- ein erster Teil enthält Angaben zur Struktur der zu importierenden Datensätze (Satzendmarken, Feldkennungen, gegebenenfalls Satzlängencodierung u. ä.), in der Beispieltabelle im Anhang sind dies die Angaben al=2, am=2, fs=5 und fc=4; die benötigten Informationen über die interne Struktur von Biblofile-MARC sind dem Handbuch zu entnehmen bzw. bei HEX-ASCII Anzeige eines Biblofile-MARC Datensatzes auf dem Bildschirm ersichtlich;
- ein zweiter Teil regelt die globalen Austauschvorgänge (Ersetzen von Sonderzeichen etc.), im Beispiel ist dies der mit „p h aouAOÜ äöüÄÖÜ“ beginnende Bereich; diese erste Anweisung ersetzt beispielsweise alle Kombinationen aus h mit einem der in der darauffolgenden Gruppe stehenden Buchstaben durch die anschließend aufgeführten deutschen Sonderzeichen;

⁶⁶ Gerade die Umformung dieses letztgenannten Dateityps ist normalerweise besonders problematisch, da eine solche Datei nicht einfach sequentiell verarbeitet werden kann.

⁶⁷ Anfängliche Inkonsistenzen beim Import bestimmter Dateien ließen sich nach langer Suche darauf zurückführen, daß das System Biblofile selbst bisweilen Daten produziert, die den eigenen formalen Definitionen nicht genügen; ob der Fehler in den auf CD vorliegenden Daten selbst oder im Verarbeitungsvorgang unter Biblofile gesucht werden muß, war allerdings nicht zu entscheiden.

⁶⁸ Der Beginn der entsprechenden Konversionstabelle MARCBF.SIM ist im Anhang wiedergegeben.

– ein dritter Teil enthält die konkreten Zuweisungsbefehle. Auch er hat eine klare Syntax: auf die Kennung des Allegro-internen Feldes (die SWB-Kategoriennummer) folgen mit „T“ gekennzeichnet die Nummer der MARC-Feldkennung (in Blöcken zu 256 zuzüglich Differenz ausgedrückt) und endlich verschiedene Bearbeitungsbefehle, von denen einige im Beispiel im Anhang erläutert sind.

Nachdem diese Tabelle erstellt und damit tatsächlich auch die Hauptarbeit getan ist, kann der Umformungsvorgang unter Allegro beginnen. Der Vorgang ist komplett dialoggesteuert, kennen muß man nur den „Importparameter“: hier ist der Name der vorher erstellten Zuordnungsdatei (im Beispiel MARCBF von MARCBF.SIM) anzugeben. Für den eigentlichen Umformungsvorgang kann der Anwender unter mehreren Optionen wählen: er kann die Umformung über eine Bildschirmanzeige der Allegro-Daten und der Originaldaten verfolgen und gegebenenfalls anhalten, um das Produkt zu redigieren (denn natürlich kann auch das beste Konvertierungsprogramm nicht beispielsweise AACR-Ansetzungen in RAK-Ansetzungen verwandeln, inhaltliche Anpassungen bleiben hier notwendig); er kann über einen komplexen Suchbegriff⁶⁹ die Menge der zu importierenden Daten eingrenzen; er kann den Import in mehrere Dateien gleichzeitig vornehmen lassen; er kann schließlich auch die Umformung völlig selbständig ohne Anzeige der Originaldaten vornehmen lassen, dann erscheint auf dem Bildschirm nur die Nummer des bearbeiteten Satzes und dessen Länge in bytes.

4.6.2. Die 'Importfunktion' von Allegro ist enorm flexibel und anpassungsfähig, die eigentliche Funktion kann auch als relativ komfortabel bezeichnet werden; einzig die Erstellung der Konvertierungstabellen bedeutet unter Umständen mehrstündige konzentrierte Arbeit, doch ist dies zum einen ein einmaliger Vorgang, zum anderen sind in der vorliegenden Version von Allegro schon eine ganze Anzahl von Konvertierungen definiert,¹⁰ die gegebenenfalls nur kopiert und an wenigen Positionen geändert werden müssen.

⁶⁹ Mittels Boolescher Operatoren, die Syntax entspricht dem Selektionsmodus, der für die Volltextsuche vorgesehen ist und weiter unten skizziert wird.

¹⁰ So beispielsweise Tabellen zur Umformung von Daten des Niedersächsischen Monographienverbundes in MAB 1, von dBASE III in MAB 1, von Daten aus WordPerfect in Daten des Niedersächsischen Monographienverbundes, von OCLC-Daten in DB-MAB etc.

4.6.3. Angesichts des oben Beschriebenen dürften die Voraussetzungen für einen später einmal zu realisierenden online-Import gut sein.

4.6.4. 10000 Datensätze wurden in einer Stunde und 17 Minuten importiert. Für die Funktionshandhabung sind 30 Sekunden hinzuzurechnen. Um die importierten Daten anschließend im Schnellzugriffsmodus nutzen zu können, ist ein (einmaliger) Indizierungslauf notwendig; dieser nimmt erhebliche Zeit in Anspruch: im Testfall waren es über fünf Stunden – allerdings bei sehr komplexer Steuerdatei und entsprechend vielen produzierten Schlüsselausdrücken.

5. Datenausgabe, Exportmodus

5.1. Zetteldruck, Bildschirmausgabe einzelner Aufnahmen:

5.1.1. Die Druckausgabe ist unter Allegro ähnlich wie im Falle der Importfunktion völlig frei gestaltbar, da sie im Prinzip dem zweiten Teil des allgemein skizzierten Umformungsvorganges entspricht. Auch hier ist über Parametrierungen in der Datenmanipulationssprache von Allegro jedes beliebige von den Formatierungsmerkmalen der Ausgangsdaten her denkbare Ergebnis erzielbar – sowohl hinsichtlich der Datenselektion als auch in Bezug auf die Darstellung.

5.1.2 Die Bildschirmausgabe des Druckformates ist möglich; sie ist eine der Standardfunktionen von Allegro.

5.1.3. Die Druckformatierung kann nur offline geschehen.

5.1.4. Allegro kann automatisch die von den RAK geforderten Nebeneintragen erzeugen und auf Zetteln ausgeben.

5.1.5. Über die obenerwähnten Parametrierungen sind beliebige Umbruchwünsche realisierbar.

5.1.6. Hier gilt sinngemäß das zur Importfunktion Gesagte: die Funktionen sind sehr komfortabel, auch die Oberfläche ist ausreichend informativ und übersichtlich, einzig die Parametrierungen als Voraussetzung sind mit erheblicher Arbeit verbunden und erfordern unter Umständen

EDV-Erfahrung, wenn auch die Probleme in der Regel einfacher zu lösen sind als im Falle der Importfunktionen.¹¹

5.2. Listendruck, Exportmodus :

5.2.1. Es waren keine Schwächen auch bei komplexen Ordnungsaufgaben festzustellen.

5.2.2. Hier gilt das unter 5.1.1. Gesagte: die Ausgabe ist völlig frei gestaltbar.

5.2.3. Der Exportmodus funktioniert analog zum Import von Daten: es wird eine Konvertierungstabelle (Kennung *.?PR) definiert, die globale Austausch- und Selektionsvorgänge festlegt und Zuweisungen von Feldinhalten auf Ausgabepositionen in einer Datei oder gedruckten Liste enthält, wobei bis zu 100 verschiedene Zwischentexte als Kommentare o. ä. an beliebige Stelle eingefügt werden können. Auch hier kann der Inhalt der Felder vor der Ausgabe mit Manipulationsbefehlen vorbehandelt werden, was sogar die nachträgliche Zerlegung von Aufnahmen mit nicht ausreichenden Differenzierungen erlaubt. Hier deuten sich interessante Perspektiven für die Verwendung dieser Funktion im Rahmen von Vorhaben zur Altbestandskonvertierung an.

Nach Definition einer solchen Tabelle müssen nur noch im Dialog die zu exportierende(n) Datei(en) ausgewählt, der Exportparameter angegeben und das Ausgabeziel (eine Datei mit Namen oder der Drucker) bestimmt werden.

¹¹ Doch gilt hier, wie auch bei den Importfunktionen, eine wohl richtige Einschätzung von Eversberg selbst: „Allegro-C' ist nicht vergleichbar 'benutzerfreundlich' wie einige andere Systeme. Dafür bietet es ein erheblich größeres Spektrum von möglichen Variationen, die der Benutzer nach seinen Erfordernissen gestalten kann. Das erfordert einiges Einarbeiten und genaues Formulieren der eigenen Wünsche und Vorstellungen. Das jedoch sollte am Anfang jeder DY-Anwendung stehen.“ (Allegro-Handbuch Abschnitt 1, S. 12; Hervorhebung in der Quelle).

5.2.4. 10000 Datensätze wurden in 53 Minuten und 40 Sekunden exportiert. Für die Funktionshandhabung sind 25 Sekunden zu veranschlagen.

5.2.5. Das Exportprodukt ist eine reine ASCII-Datei und als solche mit allen gängigen Texteditoren weiterzuverarbeiten.

5.2.6. Es gelten die unter 5.1.6. gemachten Anmerkungen.

5.2.7. Angesichts des oben Beschriebenen dürften die Voraussetzungen für einen später einmal zu realisierenden online-Export gut sein.

6. Zugriffs- und Recherchefunktionen

6.1. Schnellzugriffsmodus:

6.1.1. Der Schnellzugriff ist in Allegro seit der Version 8.6 prinzipiell neu gestaltet und gegenüber den Vorgängerversionen erheblich leistungsfähiger und komfortabler. Obwohl in den Grundfunktionen fertiggestellt, wird dieser Teil von Allegro noch ständig in puncto Komfort und Funktionenvielfalt erweitert. Es sollen daher hier nur die wichtigsten Funktionen angesprochen werden; gerade hier hält Allegro auf der Bedienungsebene (also ohne vorgängige zusätzliche Parametrisierungsarbeiten) eine ständig wachsende Vielzahl von Beeinflussungsmöglichkeiten bereit.

Der Schnellzugriff geschieht in Allegro über Indexdateien, die (vom Anwender bestimmbar) beliebige Teile von Datensätzen als Indexausdrücke enthalten können. Jede Indexdatei kann zudem eine beliebige Zusammenstellung von Allegro-Dateien (natürlich nur innerhalb einer Konfiguration) erschließen, jede Allegro-Datei kann über beliebig viele Indexdateien nach verschiedenen Schlüsseln zugriffsfähig gemacht werden.

⁷² Diese Zeiten sind stark von der Parametrierung abhängig: sie können je nach Anzahl der zu konvertierenden und auf die Platte auszugebenden Dateien nach oben und unten um ca. 25% schwanken. Die 10000 tatsächlich exportierten Sätze beschränkten sich auf 17 Ausgabefelder, die zudem alle bei 50 Zeichen abgeschnitten wurden. Auf diese Weise wurde versucht, der Importfunktion von LIPOCCLTR,OGCNZLIKOMMEN.

Die Indexdatei präsentiert sich dem Benutzer für den Zugriff als alphanumerisch geordnete Suchliste auf dem Bildschirm; in dieser Suchliste ist eine freie Bewegung mit dem Cursor, seitenweises Blättern oder auch das direkte Anspringen eines Suchbegriffs bzw. eines Alphabetabschnitts über eine für die Positionierung anzugebende Zeichenkette möglich. Bei jedem Indexbegriff ist zudem die Zahl der über ihn erreichbaren Datensätze vermerkt. Ein Datensatz kann einfach durch Betätigen der rechten Cursortaste (bzw. der Enter-Taste) vom jeweiligen Suchbegriff aus erreicht werden; anschließend kann innerhalb der durch diesen Suchbegriff erschlossenen Ergebnismenge oder in der Reihenfolge der Gesamtdatei geblättert werden. Auch ein Bearbeiten der gefundenen Datensätze, die Eingabe neuer Sätze und das Löschen von Sätzen ist an dieser Stelle möglich. Der Rücksprung in die Suchliste erfolgt einfach durch Betätigen der linken Cursortaste.

Die Bildung der Indexbegriffe wird über eine Datei vom Typ *.?PI gesteuert, die vom Anwender frei gestaltet werden kann; auch hier kommt wieder das umfangreiche Parametrisierungssystem von Allegro (mit seinen Anforderungen an den Anwender) zum Einsatz. Für die Erprobung wurde eine Steuerdatei verwendet, die eine Indexdatei mit Zugriffsmöglichkeiten über Verfasser, Körperschaftsnamen, ISBN, Titelstichwörter (über eine Stopwortliste eingrenzbar), Schlagwörter und Serientitel generierte. Doch sind beliebige andere Zugriffskombinationen generierbar: so könnte beispielweise ebenso mit 'matchcodes' gearbeitet werden; die Bildung eines Index aus den ersten vier Zeichen des Verfassernamens und den ersten vier Zeichen des Titels ist ebenso möglich wie ein aus den ersten fünf Zeichen des Verfassernamens und dem Erscheinungsjahr gebildeter Index. Als hilfreich bei der Erprobung von Steuerdateien erweist sich das mitgelieferte Zusatzprogramm „keytest“, das eine schnelle Bildschirmanzeige der in einem Indexierungslauf gebildeten Suchbegriffe erlaubt: nicht gewünschte Begriffe (bzw. formal eingrenzbare Gruppen) können anschließend noch durch Aufnahme in die Stopwortliste (bzw. durch Parametrisierungen) eliminiert werden.

6.1.2. Die Erfassung von Daten kann in Allegro im 'normalen' Redigiermodus oder direkt im 'Schnellzugriffsmodus' geschehen. Im ersteren Fall und natürlich auch nach dem Import von Daten muß nachträglich eine Indexdatei erstellt werden, was je nach Aufbau der Steuerdatei (und damit Anzahl der generierten Suchbegriffe) unterschiedlich viel Zeit in Anspruch nimmt. Im Schnellzugriffsmodus wird

die Indexdatei, unter der die Erfassung läuft, im Falle von Hinzufügungen, Löschungen oder Änderungen in indexrelevanten Feldern automatisch aktualisiert.

6.1.3. Die Indexeinträge können in verschiedener Weise rechts maskiert werden. Zum einen ist ein einfaches Trunkieren aller Register-einträge bei einer ad hoc festzulegenden Position in der Zeichenkette möglich; zum anderen kann eines der Satzzeichen „,:/“ als Endmarke benutzt werden, um etwa sämtliche Einträge unter „Schulze“ mit verschiedenen Vornamen oder alle Stücke einer gezählten Serie unter einem Suchbegriff zusammenzufassen. Beide Verfahren erfordern keine zusätzlichen Parametrisierungsarbeiten, in beiden Fällen werden die vorher unter verschiedenen Suchbegriffen geführten Ergebnismengen automatisch vereinigt.

6.1.4. 'Browsing' in einer Bildschirmliste ist die Grundtechnik des Schnellzugriffs unter Allegro.

6.1.5. Eine Präzisierung der Suche im Dialog ist auf verschiedenen Wegen möglich. Die unter verschiedenen Suchbegriffen zugriffsfähigen Ergebnismengen können mittels der Operatoren UND, ODER und NICHT direkt kombiniert werden; dies geschieht durch Voranstellen von „+“, „/“ und „-“ vor die jeweiligen Suchbegriffe in der Liste; die Bildung komplexer Suchbegriffe und die Hierarchisierung innerhalb derselben über Klammerung (wie im Volltextsuchmodus von Allegro) ist allerdings derzeit noch nicht möglich. Auch können nun die so gewonnenen Ergebnismengen (maximal 10000 Datensätze) in frei definierbare Dateien exportiert werden und stehen dann zur Weiterverarbeitung und beliebigen Kombination untereinander zur Verfügung.

6.1.6. Für die Indexgenerierung bei Erfassung und Änderung im Schnellzugriffsmodus benötigt Allegro auch bei 10000 Datensätzen nicht mehr als 5 Sekunden pro Datensatz. Für die nachträgliche Indizierung großer Datenbestände (sicher nicht der Regelfall) ist erheblich mehr Zeit zu veranschlagen; die Laufzeiten sind allerdings stark von der Komplexität der Steuerdatei abhängig. Im Testfall waren mehr als fünf Stunden für die Reindizierung von 10000 Sätzen erforderlich, das Programm zur Indexgenerierung ist jedoch für die nächste auszuliefernde Version ganz erheblich (nach Angaben von EVERSBERG um 70%) beschleunigt worden. Ein Mitstoppen der Zeit ist hier nicht erforderlich: die Tätigkeit des Programms wird in einer Protokolldatei dokumentiert.

6.2. Volltextsuche:

6.2.1. Die in der aktuellen Datei definierten Formalkategorien können wahlweise zur Präzisierung der Suche herangezogen werden.

7.1.1. Eine beliebige zeichenweise Maskierung ist möglich. Hierbei hat aber in erster Linie nur die Mittelmarkierung Bedeutung, da das Programm rein zeichenkettenorientiert sucht, ohne beispielsweise von sich aus Worttrennungen als logische Einschnitte zu behandeln (eine wortorientierte Suche ist aber dennoch möglich, der Anwender muß dann allerdings das Blankozeichen vor und nach dem Wort mit eingeben); es wird somit in der Praxis immer mit Links- und Rechtsmaskierung gesucht, auch wenn dies nicht explizit gewünscht wird.

7.1.2. Bei der Bildung komplexer Suchbegriffe können drei Gruppen von Operatoren verwendet werden:

- die bekannten logischen ('Booleschen') Operatoren (UND, ODER, NICHT),
- die Vergleichsoperatoren KLEINER und GROESSER (die je nach Folgezeichen numerisch oder lexikalisch operieren),
- Operatoren, die die Suche auf ein Feld oder eine Feldgruppe eingrenzen. Bei der Zusammensetzung der Ausdrücke können Klammern verwendet und mehrfach geschachtelt werden.

7.1.3. Das Durchsuchen der gesamten Testdatei nahm 11 Minuten in Anspruch.

6.3. Abgesehen von dem nicht sonderlich ansprechenden Laufzeitverhalten ist die Funktion zwar durchaus leistungsfähig, jedoch wenig komfortabel: so ist es allein Sache des Anwenders, syntaktisch korrekte Ausdrücke zu bilden, das System unterstützt ihn hier durch keinerlei Hilfen oder Warnungen. Positiv zu vermerken ist, daß die Suchergebnisse nicht nur in eine Allegro-Datei geschrieben und weiterverwertet, sondern auch wie im normalen Erfassungsmodus von Allegro redigiert werden können.

7. Sacherschließung

Ob eine Sacherschließung überhaupt praktiziert wird, liegt ebenso in der Hand des Benutzers wie deren Gestaltung: selbstverständlich können beliebige Sacherschließungsfelder eingerichtet werden. Eine kontrollierte Erschließung allerdings ist (wie sich schon aus den Ausführungen zur authority control unter 4.2.1. ergibt) nicht möglich. Zwar ist beispielsweise die Führung einer getrennten Thesaurusdatei denkbar, diese kann auch bei der Bearbeitung als zweite Datei im Speicher gehalten werden.⁷³ Eine Übernahme von Thesaurusdaten aus der Suchbegriffsliste ist dann ähnlich wie oben bei den Namensansetzungen geschildert möglich, doch ist weder eine echte Verknüpfung beider Dateien noch eine wirkliche Ansetzungskontrolle für die vergebenen Deskriptoren denkbar.

8. Sonstige allgemeine Kriterien

8.1. Nachdem BRANNEMANN Allegro als instabil eingeschätzt hatte⁷⁴ und auch noch die erste dem Verfasser vorliegende Version (Februar 1988) Inkonsistenzen und Unreinheiten in der Programmierung aufwies, die eine beunruhigende Zahl von 'Abstürzen' nach sich zogen, kann Allegro in der jetzt aktuellen Version als stabil bezeichnet werden. Einzig im Zusammenhang mit der Indexfunktion kam es noch vereinzelt zu Problemen. Angesichts der mit der bisherigen Entwicklungsarbeit gemachten Erfahrungen kann aber davon ausgegangen werden, daß auch diese Unreinheiten noch zu beseitigen sein werden. Allerdings ist Allegro an bestimmten Stellen⁷⁵ durch Bedienungsfehler leicht zu beeinträchtigen. In der Mehrzahl der Fälle hat dies nur einen Rücksprung in die Betriebssystemebene zur Folge, echte 'Abstürze' sind selten und betreffen auch nie die Datenbereiche, haben also auch keine Datenverluste zur Folge – dennoch wäre etwas mehr Fehlertoleranz sicher nicht fehl am Platze.

⁷³ Dieses Vorgehen skizziert Eversberg in den Allegro-news 3-88, S.2-3.

⁷⁴ BRANNEMANN (1987), S. 51.

⁷⁵ So zum Beispiel bei dem Versuch, Indexdateien ohne vorhandene Steuerdatei zu generieren (auch wenn für die Steuerdatei ein falsches Verzeichnis angegeben wird: wllrd1:) Oder b(i falscher Syntax in der Verzeichnisangabe zum Anwählen von Dnlink,11.

8.2. Zu dem differenzierten und komfortablen Schnellszugriffskonzept gibt es in Allegro ebenfalls seit der Version 8.6 erfreulicherweise auch ein Sicherungskonzept, das die Initiativen des Anwenders in diesem Bereich zumindest unterstützt. Die Datensicherung ist zwar im Prinzip immer noch ins Belieben des Anwenders gestellt, doch erfolgt eine Protokollierung aller sicherungsrelevanten Änderung in einer zur jeweiligen Datei automatisch erstellten Datei vom Typ *.LOG. Es empfiehlt sich nun, periodisch backup-Abzüge der zu einer Datenbank gehörigen Dateien (die ja alle dieselbe Buchstabenkennung vor dem Punkt haben) anzufertigen. Im Falle eines Plattenfehlers, Stromausfalls oder sonst eines Datenbankzusammenbruchs kann dann mit Hilfe des mitgelieferten Programms UPDATE.EXE die Datenbank nach dem Rückladen des Sicherungsabzuges wieder in den Zustand nach der letzten fehlerfreien Speicherung gebracht werden.⁷⁶

Allgemein kann positiv bemerkt werden, daß die transparente Struktur der Allegro-Dateien den Bedürfnissen nach Datensicherung stark entgegenkommt .

8.3.Über Kommandozeilenparameter kann bei Allegro zumindest im Ansatz ein Zugriffsschutz realisiert werden: dem Benutzer des Schnellzugriffsmoduls können Lese- und Schreibberechtigung in drei Stufen zugeteilt werden, allerdings ist dies wirklich nur ein Schutz vor einer vom Bediener nicht beabsichtigten Aktivität; zumindest innerhalb der batch-Oberfläche ist eine Änderung dieses Bedienerstatus jederzeit leicht möglich .

8.4. Allegro verfügt über einige wenige vom Benutzer aktivierbare Hilfsbildschirme im Zusammenhang mit Dateneingabe und Schnellzugriffsmodus . Die hierin und in den automatisch eingeblendeten Bearbeitungsleisten enthaltenen Informationen sind erst bei guter Einarbeitung ausreichend. Es sind aber beliebige weitere Hilfsschirme erstellbar.

Eine Besonderheit von Allegro ist auch die Möglichkeit, sämtliche Systemmeldungen durch einfache Textvariationen in den Dateien `UIF*.*` ändern zu können, wenn die mitgelieferten Texte zu wenig informativ oder unklar erscheinen.

⁷⁶ Mit einer im Handbuch noch nicht beschriebenen Kommandooption von UPDATE soll es nach Angaben des Entwicklers übrigens auch möglich sein, eigene Aufnahmen durch zuvor importierte Fremdaufnahmen ersetzen zu lassen.

Das begleitende Handbuch ist bei der Arbeit mit Allegro eine große Hilfe, wenn auch die Präsentation der Informationen nicht immer ganz übersichtlich ist. Die Qualität der Informationen hingegen ist durchweg gut: auf die Angaben ist durchweg Verlaß und die Erklärungen sind da, wo sie wirklich notwendig sind (also im Bereich des komplexen Parametrierungssystems), ausreichend. In einem fällt der Text besonders angenehm auf: er weist durchaus auch auf Unfertigkeiten und Desiderate im System hin, beschönigt nichts und macht keine Versprechungen, die von Allegro nicht einzulösen sind. Angesichts der Vielzahl neu hinzugekommener Teilfunktionen und Optionen wird auch diese Fassung des Handbuchs (Stand Oktober 1988) wieder gründlich zu ergänzen und zu überarbeiten sein – sehr wünschenswert wäre dann ein Befehls- und Funktionsindex: zur Zeit ist das Auffinden einer Einzelinformation im Handbuch eine mitunter sehr mühselige Angelegenheit.

8.5. In Niedersachsen kann Allegro an Bibliotheken in Landesträgerschaft frei abgegeben werden. Ansonsten sind der Preis und die Vertriebsbedingungen durch die Firma DABIS noch nicht ganz geklärt, zumal die Zustimmung des zuständigen niedersächsischen Landesministeriums noch aussteht. Voraussichtlich wird Allegro zu einem Preis von 1500-2000 DM abgegeben werden. Im Lieferumfang werden dann vermutlich auch Updates und Systempflege für ein Jahr nach Lieferung enthalten sein, auch Campuslizenzen wird es geben: bei Kauf von Allegro durch die zentrale Bibliothek kann das Paket dann im gesamten universitären Bibliothekssystem installiert werden; allerdings besteht hier noch keine preisliche Klarheit.

E zusammenfassende Charakterisierung

Allegro ist ein weitgehend ausgereiftes, allerdings nur begrenzt anwenderfreundliches System vor allem zur Erfassung und zur Ausgabe bibliographischer Daten. Positiv besonders herauszuheben sind seine enorme Flexibilität, der inzwischen ausgereifte und komfortable Schnellsuchmodus, der hohen bibliographischen Differenzierungsgrad und nicht zuletzt der vergleichsweise niedrige Preis. Zu beachten ist dabei allerdings, daß eine erhebliche Einarbeitungszeit eingeplant werden muß, bevor die Stärken der Parametrierungsmöglichkeiten von Allegro wirklich genutzt werden können, auch sind hier EDV-Kenntnisse nicht fehl am Platze. Doch kann dieser Aspekt

auch positiv gewertet werden: Allegro zwingt den Anwender zu einer sehr intensiven und präzisen Systemanalyse und bietet keine nur oberflächlich besehen akzeptablen Scheinlösungen.

Vom Laufzeitverhalten und von der Benutzerfreundlichkeit her ist auch der Schnelzugriffmodus von Allegro inzwischen sehr ansprechend: in der nun vorliegenden Form erscheint der Einsatz des Schnelzugriffsmodus als Rechercheoberfläche für den Benutzer durchaus möglich;⁷⁷ dagegen spricht allerdings der nicht wirklich realisierte Zugriffsschutz, der Benutzer könnte in Allegro jederzeit beliebige Änderungen im Datenbestand vornehmen, so daß eine Benutzerrecherche allenfalls in einem speziell für diese Zwecke bereitgestellten Datenbankabzug denkbar ist (der dann entsprechend Platz benötigen würde). Unverändert karg und wenig ansprechend, daher für die Benutzerrecherche auch weniger geeignet ist die Freitextsuche unter Allegro.

Ein gravierender Mangel von Allegro ist unverändert dessen Eindateienstruktur, die eine echte Erfassungskontrolle über Stammdateien unmöglich macht. Realistischerweise kann mit einer völligen Umstellung der Dateienstruktur von Allegro zumindest in naher Zukunft wohl nicht gerechnet werden.

Gesondert herauszuheben sind die Offenheit, Transparenz und Portabilität von Allegro. Diese Faktoren geben dem potentiellen Anwender zusammen mit der enormen Anpassungsfähigkeit der Funktionen und Dateien eine gewisse Sicherheit, mit dem Einsatz von Allegro nicht in eine elektronische Sackgasse zu geraten.

Dem System Allegro ist seine Herkunft aus der bibliothekarischen Praxis deutlich anzusehen: die von dort herrührenden Eigenschaften machen es zu einem guten Instrument für die Erfassung, Bearbeitung und Ausgabe bibliographischer Daten in den Händen einer eingearbeiteten Fachkraft.

⁷⁷ In früheren Versionen war daran kaum zu denken.

3.2. LIDOS 3.0

A Charakterisierung

Das Literaturdokumentationssystem LIDOS kann nur mit Einschränkungen überhaupt in diese Untersuchung einbezogen werden: tatsächlich markiert es – wie sein Name schon andeutet – zumindest die Grenze des bibliothekarischen und von den Eigenheiten gewachsener lokaler Bibliothekssysteme erheblich beeinflussten Anwendungsbereiches gegenüber dokumentarischen, weit eher auf die inhaltliche Erschließung auch unselbständiger Literatur abzielenden Anwendungen. Vieles spricht dafür, daß LIDOS in seiner gegenwärtigen Form sich aus bibliothekarischer Sicht schon jenseits dieser Grenze befindet; einige dieser Punkte werden in der Beschreibung anhand der Kriterienliste offenkundig. Vor allem aber entspricht diese Einschätzung derjenigen der Entwickler selbst: Franz-Josef Land, selbst an der Systementwicklung beteiligt und zugleich Anwender, gab auf einer Vorführung des Systems am 15.03.1988 im Rechenzentrum der Universität Bonn an, das Produkt in seiner gegenwärtigen Form sei für Anwender in Bibliotheken der Universität Bonn wenig geeignet. Als Katalogisierungssoftware wolle er LIDOS für die Bonner Verhältnisse nur ungern verkaufen, da die dann notwendig anfallenden Anpassungswünsche der Anwender der Firma sicher mehr Arbeit bescherten würden als im Verhältnis zu dem dabei zu erzielenden Erlös wünschenswert sei. Es sei zwar prinzipiell möglich, LIDOS an diese bibliothekarischen Erfordernisse anzupassen, jedoch sei dann eine umfangreichere Entwicklungsarbeit notwendig, die entsprechend honoriert werden müsse.

Diese klare und wohlüberlegte Aussage ist umsomehr zu begrüßen, als leider gerade aus Bibliothekskreisen Stimmen laut geworden sind, die den Einsatz von LIDOS als bibliothekarisches Instrument ernsthaft ins Auge fassen. Ist dies, wie etwa im Falle der Planungen im Rahmen des Bonner Bibliothekssystems, aus der isolierten und mit Blick auf die Erfordernisse des Gesamtsystems unter Umständen auch reduzierten Sicht von Institutsanwendern immerhin noch verständlich (wenn auch aus der Sicht der Zentralbibliothek nicht unbedingt akzeptabel), so ist es denn doch erstaunlich, wie unkritisch in der ja durchaus einflussreichen Y.cit'schrift ABI-Technik LIDOS generell als „Arbcin,mittel

für den bibliothekarischen Alltag" auch an „großen Bibliotheken aller Typen oder auch im Stadtbüchereisektor" angepriesen wird.⁷⁸ Wenn sich zudem in diesem Beitrag nicht als Zitate ausgewiesene Formulierungen der Broschüre zur Demo-Diskette wiederfinden,⁷⁹ so ist den dort gemachten Aussagen gegenüber denn doch erhebliche Skepsis geboten. Ein weiteres Eingehen auf die Schwächen dieses Beitrages ist hier jedoch nicht am Platze: der Leser möge selbst die gegebene Beschreibung mit dem Beitrag von STRZOLKA vergleichen, wobei zudem zu berücksichtigen ist, daß STRZOLKA die (wesentlich weniger leistungsfähige) Version 2.0 von LIDOS untersucht hat.

LIDOS ist von seiner Grundkonzeption her ein vor allem hinsichtlich seiner Sacherschließungsmöglichkeiten hochentwickeltes System zur Verwaltung auch unselbständiger Literatur. Ein durchdachtes, übersichtliches und sehr differenziertes Menusystem gestaltet die Arbeit mit LIDOS äußerst komfortabel, der Anwender muß so gut wie keine Kommandos und Dateinamen im Kopf behalten. Nahezu sämtliche Funktionsaufrufe werden über Funktionstasten getätigt, einige von ihnen haben zudem in allen Anwendungsbereichen dieselbe Funktion: so führt etwa die Taste F1 immer in das vorhergehende Menu zurück, F2 leistet dasselbe und ignoriert dabei alle vorgenommenen Veränderungen, F9 sichert in allen Eingabebereichen das Arbeitsergebnis. Da das System zudem das Springen zwischen unterschiedlichen Menubereichen ermöglicht, entfällt der bei anderen Programmen von geübten Anwendern oft als unangenehm empfundene Tempoverlust bei konsequenter Menuführung weitgehend.

Der gesamte Programmcode von LIDOS ist in einer Datei (LID3PRM.COM) enthalten, die von einer Startdatei (LIDOS.EXE) aufgerufen wird. Eine kurze Datei (LIDOS.PIF) enthält Informationen zur Umgebung des Systems, die beiden Dateien LID2LID3.EXE und LID2LID3.PRg sind für die Umwandlung von mit LIDOS 2.0 erfaßten Daten in Dateien unter LIDOS 3.0 zuständig. Bis auf die

⁷⁸ Die Rede ist von dem Beitrag von STRZOLKA; die Zitate befinden sich auf S. 59.

⁷⁹ Als Beispiel seien nur die von STRZOLKA bei anderen Programmen monierten „barocken Schnörkel zur Verzierung" (S. 60) genannt, die eben auch die Kurzbeschreibung zur Demo-Diskette (LAND 1987b) mit exakt dieser Formulierung auf S. 5 ablehnt.

Möglichkeit, die Starttexte zu verändern, kann die Funktionsweise dieses Systemkerns vom Anwender nicht beeinflußt werden. Mitgeliefert werden weiter eine große Zahl von Druckertreibern (*.OOL) und Ausgabeformaten (*.NNL), Beispieldateien und Beispielprogramme zur Überführung von Fremddaten in das Importformat von LIDOS. Wer diese Umformungsprogramme nicht selbst schreiben kann (oder möchte), kann zusätzlich das Modul DOWNLOAD.EXE zur Überführung von Fremddateien in das LIDOS-Importformat erwerben.

B Entwicklung

LIDOS wird von der Firma „Angewandte Statistik und Software-Entwicklung Doris Land" so entwickelt vertrieben. Es gibt inzwischen 18 Vertriebsbeauftragte vor Ort. Der zügige Aufbau des Vertriebsnetzes und die offensichtlich gute Anwenderunterstützung (auch durch die zweimonatlich erscheinenden „LIDOS-News" mit vielen praktischen Tips und Anregungen) lassen den Schluß zu, daß auch mittelfristig mit Unterstützung und Weiterentwicklung von LIDOS durch die Herstellerfirma gerechnet werden kann.

Das Programm ist seit Mitte 1987 auf dem Markt, sein Vorgänger LIDOS 2.0 seit Anfang 1986. Es ist in der dem Verfasser vorliegenden Form fertig entwickelt, alle Funktionen stehen (zumindest theoretisch) voll zur Verfügung.

Die Referenzliste von LIDOS 2.0 verzeichnet eine Auswahl von Anwendern, die bis auf eine Ausnahme³ als eigenständige Institutionen selbständig und mit isolierten Datenbeständen arbeiten.

Die Vertreiberfirma liefert auf Bestellung eine Demo-Diskette (der Preis von 30 DM wird beim Kauf angerechnet), auch das Handbuch ist dort zum Preis von 130 DM gesondert erhältlich.

³⁰ Postfach 1126, 8507 Oberasbach.

³¹ Der Vertrieb befand sich bis 1988 bei der Fa. EXpress-Edition in Berlin.

³² Erhältlich beim Entwickler.

³³ Cloo1r11ph1NthcN Institut der Universität Bern.

C Literatur

Die Version 2.0 von LIDOS war Gegenstand von Besprechungen unter anderem in BuchMarkt (BRUNOLD) und ABI-Technik (STRZOLKA), ein Vergleich von LIDOS mit Allegro-C findet sich in den Beiträgen EVERSBERG (1986c) und EVERSBERG (1987a).

Die Version 3.0 wurde in der Zeitschrift PC Magazin vorgestellt (der Aufsatz von KOST).

Es existiert ein ausführliches Handbuch (LAND 1987a) und eine Kurzeinführung zur Demodiskette (LAND 1987b).

D Beschreibung

1. Hardwareseitige und allgemein EDV-technische Bedingungen

1.1. LIDOS ist auf allen zum IBM-PC/XT oder AT kompatiblen Rechnern lauffähig. Es existiert auch eine Version für den Atari-ST mit geringfügig eingeschränktem Funktionsumfang.

1.2. LIDOS benötigt mindestens 256 KB Hauptspeicher, ist also auch bei extensiver Speicherplatzbelegung durch residente Programme lauffähig; die festgestellten Fehlfunktionen von LIDOS sind mit Sicherheit nicht auf die gleichzeitige Präsenz speicherresidenter Programme zurückzuführen: die entsprechenden Tests wurden mit und ohne speicherresidente Programme durchgeführt. Der Einsatz einer RAM-Disk dürfte den Programmablauf nur dann beschleunigen, wenn diese groß genug ist, um alle Dateien aufzunehmen, in denen Daten gespeichert werden; hierfür sind mindestens 200 KB erforderlich.

1.3. LIDOS ist voll mehrplatz- und netzwerkfähig.

1.4. LIDOS ist derzeit unter MS-DOS (ab Version 2.0) und dem neuen Betriebssystem OS/2 verfügbar, eine UNIX-Version ist nicht geplant.

1.5. LIDOS ist in Assemblersprache programmiert, mithin nicht portabel.

1.6. LIDOS präsentiert sich dem Anwender als kompakter Programmblock in Assembler, in dem keine Modifikationen vorgenommen werden können. Beim Erstellen einer Dokumentation erzeugt LIDOS automatisch acht Dateien vom Typ *.LD, von denen die vier Dateien mit Zahlenwerten vor dem Trennpunkt des Programmnamens die Daten sowie offensichtlich jeweils sämtliche Systemmeldungen und Teile des Programmcodes, die anderen vier Dateien Strukturinformationen zur betreffenden Datei enthalten.

Diese wenig durchsichtige Architektur macht eine LIDOS-Datenbank für den Anwender auf der Ebene des Betriebssystems weitgehend undurchschaubar, Dateioperationen (so etwa das Ineinanderkopieren von Dateien) sind kaum denkbar, auch die Datensicherung wird durch diese unübersichtliche Struktur nicht eben vereinfacht.

1.7. Eine Online-Schnittstelle für den Datenaustausch existiert nicht und ist auch nicht vorgesehen.

2. Datenbanktechnische Spezifikationen

2.1. Grenzwerte:

2.1.1. Die Anzahl der Datensätze pro Dokumentation ist nur durch die Kapazität des Massenspeichers begrenzt; alle gemeinsam zu verwaltenden Dateien müssen sich in einer Dokumentation befinden.

2.1.2. Ein Datensatz kann maximal 64 000 Zeichen lang werden.

2.1.3. Die absolute Feldzahl ist auf 205 pro Dokumentation beschränkt (die fünf Standardtextfelder und die maximal zusätzlich einzurichtenden 200 Textfelder).

2.1.4. Die Feldlänge ist nur durch die Längenbegrenzung für den gesamten Satz eingeschränkt.

2.1.5. LIDOS kann nach maximal drei frei wählbaren Datenfeldern gleichartig benannt werden.

2.2. LIDOS verwendet eine eigene, speziell entwickelte Datenbankstruktur. Die physische Organisation der Daten ist, wie oben angedeutet, undurchsichtig, doch sind offensichtlich Datensatz und Titelaufnahme (=Dokument) keine Einheit: LIDOS legt verschiedene Elemente eines Dokuments in den Dateien *00.LD bis *03.LD der jeweiligen Dokumentation getrennt ab, wobei auch die schnellzugriffsrelevanten Elemente getrennt verwaltet werden. Ein Schnellzugriff ist über den Autor, das Jahr und das Deskriptorenfeld möglich, der Inhalt aller anderen Felder, der sogenannten 'Textfelder' ist nur im Volltextmodus recherchierbar.

2.3. Feld- und Satzlängen sind im Rahmen der Grenzwerte flexibel.

2.4. Zusätzlich zu den vier standardmäßig vorgegebenen Feldern (Autor, Co-Autor, Titel, Jahr und Deskriptor) sind bis zu 200 Felder frei definierbar und können beliebig benannt werden.

2.5. Die Feldlängen sind nicht determinierbar.

2.6. Lidos nützt den Massenspeicherplatz weitgehend ökonomisch aus; allerdings wird, wie oben dargestellt, auch bei der Bildung kleiner Dateien automatisch eine gehörige Grundportion Speicherplatz für den Aufbau der Speicherdateien verwendet, die über die aufzunehmenden Daten hinaus ca. 50 000 bytes Programm- und Hilfscode enthalten.

3. Regelwerks- und Kategorienanforderungen

3.1. Eine Strukturierung der Datensätze nach dem von MÜNNICH erstellten Schema ist theoretisch möglich, ergonomisch jedoch kaum vertretbar.

3.2. Hierarchisch strukturierte Aufnahmen sind nicht möglich.

3.3. Es existieren keine formalen Verknüpfungsmöglichkeiten.

4. Dateneingabe, Importfunktion

4.1. Ergonomie, Akzeptanz:

4.1.1. Das Erscheinungsbild und die Anordnung der Eingabefelder sind sehr ungünstig: die Felder werden in einer Kolonne untereinander angeordnet, was bei komplexen Aufnahmeformaten zu einer Verteilung auf mehrere Bildschirmseiten führt. Da zudem von der Überschrift eines jeden Feldes eine Bildschirmzeile beansprucht wird und alle Felder, auch die nicht benötigten, angezeigt werden müssen, ist eine völlig unübersichtliche, ergonomisch unsinnige Eingabemaske die Folge.

4.1.2. Die Möglichkeiten der Maskengestaltung sind minimal: es kann nur die Reihenfolge der Textfelder bestimmt werden, diese müssen jedoch als Block zwischen den obligaten Feldern für Autor, Co-Autor, Titel und Jahr am Anfang und dem Deskriptorenfeld am Schluß stehen. Darüberhinaus existieren keine Möglichkeiten der Maskengestaltung.

4.1.3. Möglichkeiten der Datenübernahme aus anderen Sätzen derselben Dokumentation bestehen nicht.

4.1.4. LIDOS verfügt über beeindruckende Editorfunktionen für die Texteingabe, die an den Komfort einer durchschnittlichen Textverarbeitungssoftware heranreichen, und die deshalb hier nicht aufgezählt werden können. Neben solchen Leistungen bis hin zu Blockfunktionen sind noch zwei Funktionen besonders hervorzuheben:

– häufig wiederkehrende Eingabeelemente können in einem Kurztextarchiv zur Mehrfachverwendung abgelegt werden und sind sehr komfortabel abrufbar;

– eine schnelle und durchaus effiziente Doublettenprüfung hilft beim Verhindern von Doppeleingaben.

4.2. Routinen zur Sicherung der Datenhomogenität:

4.2.1. Authority control ist bei LIDOS nur in der Sacherschließung über den Thesaurus möglich.

4.2.2. Plausibilitätskontrollen existieren nicht und sind auch nicht

4.2.3. Es gibt bei der manuellen Eingabe keine Unterscheidung nach Pflichtkategorien und fakultativen Eingabebereichen, nur im Autorenfeld ist eine Eingabe zwingend erforderlich – auch dann, wenn das betreffende Werk keinen Urheber hat: dann erfolgt im Autorenfeld eine Eingabe der Körperschaft oder des ersten ordnenden Elements des Sachtitels bzw. eines anderen Ansetzungselements, diese Eingabe wird durch führendes Hochkomma von einer echten Autorenangabe unterschieden.

4.2.4. Regelwerkshilfen können über die Kommentare realisiert werden, die zu allen Feldern einer Dokumentation definiert und aus dem Erfassungsvorgang heraus leicht abgerufen werden können.

4.3. Die automatische Mehrfachbelegung von Datenfeldern ist nicht möglich: ist eine Mehrfachbelegung erwünscht, müssen entsprechend zusätzliche Felder mit der gleichen Bezeichnung definiert werden, was die in 4.1.1. angesprochenen ergonomischen Unverträglichkeiten noch steigert, denn natürlich erscheinen auch diese möglicherweise bei 100 Aufnahmen nur einmal benötigten Mehrfachfelder bei jeder Aufnahme in der Eingabemaske.

4.4. Es ist der Zeichensatz des jeweiligen Rechners voll verfügbar. Möglichkeiten der Einschränkung gibt es nicht.

4.5. Auch bei größeren Datenmengen dauerte das Hinzufügen eines neuen Datensatzes nicht erheblich länger.

4.6. Importfunktionen:

4.6.1. Den Import von Fremddaten nach LIDOS empfand schon die Besprechung im PC-Magazin als problematisch. Das Verfahren ist kompliziert, teilweise unbrauchbar, funktioniert in einigen Fällen überhaupt nicht und kann dann die allergrößten Unannehmlichkeiten bereiten: Nicht zu behebbende Fehler in der Dateizuordnung unter MS-DOS, Unlesbarkeit der Command-Datei und andere Widrigkeiten waren mehrfach die Folge von Testläufen mit der Import-Funktion von LIDOS.

Im Prinzip funktioniert der Datenimport folgendermaßen: Die Übernahme von Dokumenten aus einer Datei ist (theoretisch) immer dann auch bei abweichender Dokumentenstruktur möglich, wenn die Daten als reine ASCII-Datei im Format nach der Norm DIN 1506 (ISO 2709, „Format für den Austausch von bibliographischen Daten“) in der Vorzugsform vorliegen. Daß dies in der praktischen Anwendung kaum einmal der Fall sein dürfte, haben die Entwickler wohl selbst erkannt; sie

erläutern daher den recht komplexen Aufbau dieses Formates im Handbuch korrekt und verständlich und verweisen ansonsten den Anwender auf zwei Beispielprogramme in BASIC und PASCAL, die (sehr einfach strukturierte) Fremddaten in das Format nach DIN 1506 überführen. In der ebenfalls zutreffenden Annahme, daß Programmierkenntnisse der Art, wie sie zur Erstellung analoger Programme erforderlich sind, in der Regel nicht vorausgesetzt werden können, wird der Anwender ansonsten auf das für 490 DM zusätzlich zu erwerbende Modul „LIDOS-download“ verwiesen.

Dieses Zusatzmodul soll nach den gerade in diesem Bereich leider nicht immer ganz präzisen und zuweilen falschen Handbuchangaben in der Lage sein, beliebige Fremddaten in das für LIDOS lesbare Format nach DIN 1506 zu überführen, wenn vier Voraussetzungen erfüllt sind:

- die Daten müssen als MS-DOS Datei im ASCII-Format vorliegen;
- sie müssen formatiert sein (d.h. die Felder müssen entweder durch eindeutige Feldkennungen am Anfang oder durch ihre Reihenfolge identifizierbar sein);
- alle Zeilen der Fremddaten müssen mit der Befehlsfolge „Carriage-return+Line-feed“ (ASCII 10 und 13/0A und OD im hexadezimalen Format) abgeschlossen sein;
- durch Übertragungsfehler gegebenenfalls entstehende überlange Zeilen dürfen nicht länger als 255 Zeichen sein.

Zusätzlich zu diesen Bedingungen benötigt das Modul eine für das Fremdformat erstellte Feldzuordnungstabelle, sollte keine der mitgelieferten Tabellen (Dateien mit der Kennung DWN) passend sein, ist es leicht möglich, eine solche Tabelle selbst zu erstellen, denn die Syntax des dabei anzuwendenden Befehls ist denkbar einfach: es wird zuerst die exakte Kennung des Fremdtexfeldes und anschließend durch einen Schrägstrich von diesem getrennt der Name des zugewiesenen LIDOS-Textfeldes eingegeben.

Auf diese Weise ist eine immerhin theoretisch akzeptable Prozedur geschaffen, um Fremddaten in das Format nach DIN 1506 zu überführen – leider erwies sich aber schon dieser (den eigentlichen Import ja erst nur vorbereitende) Schritt als problematisch. Dies hatte seine Ursache einerseits in den Eigenheiten der von Bibliofile erstellten

Datens⁴. Doch erwies sich die Funktion LIDOS-download zudem als äußerst anfällig gegenüber Schreib-/Lesefehlern besonders bei großen Datenmengen, so daß die ursprünglich für den Datenimport ins Auge gefaßte Vorgehensweise nicht realisierbar schien. Statt dessen wurde ein Umweg über Allegro-C gewählt: Da dieses System einerseits die Bibfile-Daten in ihrer Rohform importieren, andererseits im Exportmodus Daten produzieren kann, die exakt den von LIDOS-download geforderten Bedingungen entsprechen, wurden nun Allegro-Exportdaten mit LIDOS-download bearbeitet, und zwar in kleinen Paketen à 400 bis 500 Sätzen, die dann zumindest für LIDOS-download auch kein Problem mehr darstellten.

Die nun im Format nach DIN 1506 vorliegende LIDOS-Importdatei kann direkt mit dem Importmodus von LIDOS weiterverarbeitet werden. Hierbei werden die einzulesenden Daten zuerst in einem Vorlauf daraufhin überprüft, ob sie im Sinne von LIDOS syntaktisch korrekt und mit der Feldstruktur der Zieldokumentation kongruent sind. Es sind dabei sehr weitgehende und komfortable manuelle Nachbesserungen und Zuordnungen möglich. Wird die anschließend auf dem Bildschirm erscheinende Frage „Dokumente einspielen?“ mit „Ja“ beantwortet, werden die Daten in die gewählte Zielfeld überführt. Doch leider erwies sich diese Funktion als extrem fehleranfällig. So war es zum Beispiel möglich, daß ein- und dieselbe Importdatei, die den Prüflauf anstandslos passiert hatte, einmal mit allen 415 Sätzen eingespielt wurde, ein anderes Mal wieder wurden nur 379 Sätze eingespielt, in einem dritten Fall endlich traten im Verlauf des Einspielens Fehlfunktionen auf, die sich in die Dateizuordnungstabelle von MS-DOS fortsetzten, welche daraufhin reorganisiert werden mußte; größere Mengen importierter Daten hatten außerdem Integritätsfehler im LIDOS-Suchbaum zur Folge, die ab ca. 2 500 Datensätzen unter LIDOS in zwei von drei Fällen mit dem Systemfehler 75 quittiert wurden, der eine Reorganisation der Dokumentation verlangt; auch diese Reorganisation brachte aber leider nicht das gewünschte Ergebnis, zerstörte vielmehr die Datei COMMAND.COM auf der Festplatte und richtete auch sonst große Verwirrung in der Dateizuordnungstabelle an, die zu einem Neuformatieren der gesamten Festplatte zwangen.

⁸⁴ So standen etwa –US-amerikanischen Gepflogenheiten entsprechend –Carriage-Return und Line-Feed am Zeilenende in umgekehrter Reihenfolge und waren damit für das Programm nicht identifizierbar.

Diese Fehlfunktionen sind mit größter Wahrscheinlichkeit nicht hardwareabhängig. Mit Sicherheit sind sie nicht auf die Ausgangsdaten zurückzuführen, die anschließend nochmals sehr gründlich analysiert wurden, und deren Konsistenz ja auch durch den unter Allegro-C möglichen Import belegt ist. Sie hatten jedenfalls zur Folge, daß der Import der Testdatei nach LIDOS nicht möglich war, auch der mehrfache Import ein- und derselben Datei war nicht zu bewerkstelligen (auf diese Weise wäre das Laufzeitverhalten bei 10000 Titeln zumindest zu simulieren gewesen).

4.6.2. Abgesehen von den oben angesprochenen Fehlfunktionen ist die eigentliche Import-Funktion von LIDOS durchaus komfortabel. Da der Import nach LIDOS jedoch realistischerweise als Summe der Vorgänge unter LIDOS-download, der Importfunktion und gegebenenfalls verschiedener Umwege über Umformungsprogramme und Texteditoren gesehen werden muß, ist der 'Import' eines einzelnen bibliographischen Datensatzes allemal bequemer durch manuelle Eingabe zu bewerkstelligen als mit dem oben beschriebenen Verfahren. Doch auch größere Datenmengen sind so nur unter erheblichen Mühen nach LIDOS zu transportieren. Die Flexibilität des eigentlichen Importmodus ist sehr gering, sie wird auch durch die relativ einfach zu handhabenden Feldzuordnungen unter LIDOS-download und die Möglichkeit der nachträglichen Zuordnung im Prüflauf nicht entscheidend erhöht: die Hauptschwachstelle ist hier die Fixierung auf das Format nach DIN 1506 an der Datenschnittstelle.

4.6.3. Die Voraussetzungen für einen online-Import sind angesichts des oben Beschriebenen minimal.

4.6.4. Laufzeiten konnten nicht ermittelt werden. Legt man allerdings die Zeiten zugrunde, die in der beschriebenen Weise für den Import von 100 Datensätzen notwendig waren und rechnet diese hoch, so ergeben sich theoretisch einige Tage Arbeitszeit.

5. Datenausgabe, Exportmodus

5.1. Zetteldruck, Bildschirmausgabe einzelner Aufnahmen:

5.1.1. Die Druckausgabe ist unter LIDOS in allen nur erdenklichen Formen (Listen- und Zetteldruck) über die Erstellung sogenannter Druckformat-Formulare völlig frei gestaltbar.

5.1.2. Die Bildschirmausgabe des Druckformates ist nicht möglich.

5.1.3. Die Druckformatierung kann nur offline geschehen.

5.1.4. LIDOS kann Nebeneintragen nur erzeugen, wenn die entsprechenden Aufnahmeelemente, durch ein Hochkomma eingeleitet, im Feld „Co-Autor“ eingegeben wurden. Eine automatische Erzeugung der von den RAK geforderten Nebeneintragen ist somit kaum möglich.

5.1.5. Der Umbruch bei langen Sätzen ist möglich.

5.1.6. Die Steuerung des Drucks über die zudem jederzeit wiederverwendbaren Druckformattabellen ist sehr komfortabel, auch die Erstellung der Tabellen selbst ist vergleichsweise unproblematisch. Allerdings können die Feldinhalte leider nicht für die Ausgabe vorbehandelt werden: sie können nur unverändert ausgegeben werden (mit Ausnahme des Autorennamens).

5.2. Listendruck, Exportmodus:

5.2.1. Die Ordnungsfunktionen bei gereihter Datenausgabe sind, wie oben beschrieben, durch maximal drei Felder gleichzeitig zu steuern. Dies wird bei großen Datenmengen nicht immer ausreichen: es führt immer dann zu Fehlern, wenn beispielsweise Werke eines Verfassers in unterschiedlichen Ausgaben im gleichen Jahr erschienen sind. Des weiteren müssen bei der Einordnung zu übergehende Elemente der Aufnahme manuell durch ein Nichtsortiertzeichen gekennzeichnet werden, eine automatische Hilfe etwa durch Stopwortlisten ist nicht vorhanden.

5.2.2. Hier gilt das unter 5.1.1. Gesagte: die Ausgabe ist völlig frei gestaltbar.

5.2.3. Für den Export sind bei LIDOS zwei Fälle unterscheidbar. Zum einen ist die Ausgabe einer Liste natürlich mit analogen Gestaltungsmöglichkeiten auch in eine ASCII-Datei zur Weiterverwendung mit einer Textverarbeitungssoftware anstelle der Druckerausgabe möglich. Zusätzlich kann aber das Ergebnis der Recherche in einer LIDOS-Dokumentation im LIDOS-Ausgabeformat exportiert werden – die Gestaltungs- und Selektionsmöglichkeiten sind dabei allerdings minimal: es können nur wahlweise die Deskriptoren und sämtliche Textfelder als Gruppe berücksichtigt oder ausgeschlossen werden.

...4. Es konnten keine Laufzeiten ermittelt werden.

5.2.5. Das Exportprodukt ist eine reine ASCII-Datei, die mit jeder Textverarbeitungssoftware problemlos zu bearbeiten ist.

5.2.6. Es gelten die unter 5.1.6. gemachten Anmerkungen.

5.2.7. Die Voraussetzungen für einen Online-export sind minimal.

6. Zugriffs- und Recherchefunktionen

6.1. Schnellzugriffsmodus:

6.1.1. Der Schnellzugriff kann über den Autor, das Jahr und die Deskriptoren in beliebiger Kombination erfolgen, wobei die Zugriffslogik starr und vom Anwender nicht beeinflussbar ist.

6.1.2. Die Reorganisation der für den Schnellzugriff benötigten Sekundärdateien geschieht bei LIDOS ohne Zutun des Anwenders während der Erfassung bzw. Änderung von Dokumenten. Beim Aufbau großer Datenbestände kann gelegentlich eine vom Anwender über das Dienstmenu zu veranlassende Reorganisation der Dateien notwendig werden; hier sind unangenehme Überraschungen angesichts der oben referierten Erfahrungen bei umfangreichen Importvorgängen nicht ausgeschlossen, eine Sicherung der Daten ist vor der Reorganisation dringend geboten (das System legt dies durch eine entsprechende Warnmeldung selbst nahe).

6.1.3. Bei der Autorenrecherche ist Mitte- und Rechtsmaskierung möglich, bei der Deskriptorenrecherche ist durch die Bindung an das Thesaurusprinzip keine Maskierung möglich.

6.1.4. Das Ergebnis einer Suche kann als Autoren- oder Titelliste angezeigt werden, in der dann jeweils sehr komfortables 'browsing' mittels Cursorbewegung möglich ist, das jeweils an der Cursorposition befindliche Dokument kann durch Druck einer Funktionstaste angezeigt werden – allerdings im (unübersichtlichen) Aufnahmeformat von LIDOS.

6.1.5. Eine nachträgliche Präzisierung der Suche im Dialog ist möglich.

1.1.(), 1.n1fzoilcn konnten nicht ermittelt werden.

6.2. Volltextsuche:

6.2.1. Die Eingrenzung über die in der Datei definierten Formalkategorien (=Textfelder) ist nicht nur möglich sondern unumgänglich: die Volltextsuche unter LIDOS ist keine Freitextsuche, die den gesamten Datensatz beliebig nach Zeichenketten durchsucht, es können Zeichenketten nur feldorientiert gesucht werden. Eine globale Suche muß gegebenenfalls sehr aufwendig als kombinierte Suche in allen Standard- und Textfeldern der Dokumentation (mit Verknüpfung über die 'oder'-Bedingung) definiert werden.

6.2.2. Bei der Volltextsuche ist beliebige Maskierung möglich.

6.2.3. Es können unter Verwendung der Operatoren UND, ODER und NICHT sowie der Klammern beliebig komplexe Suchbegriffe formuliert werden; diese sind zudem mit der Autoren- und Deskriptorenrecherche verknüpfbar!

6.2.4. Laufzeiten konnten nicht ermittelt werden.

6.3. Die Recherchefunktionen von LIDOS sind sowohl hinsichtlich der wahrscheinlichen Laufzeit als auch wegen ihrer einfachen Handhabung, der Möglichkeiten der Speicherung und Weiterverarbeitung, vor allem aber deshalb sehr positiv zu bewerten, weil Suchergebnisse auch nachträglich beliebig kombiniert oder geschnitten werden können. Auch die graphische Unterstützung bei der Erstellung syntaktisch korrekter Suchaufträge fällt sehr angenehm auf.

7. Möglichkeiten der Sacherschließung

Die Sacherschließungsmöglichkeiten sind die eigentliche Stärke von LIDOS. Es können zu jeder Dokumentation bis zu 65 000 Schlagwörter in einem Register verwaltet oder eine entsprechende Anzahl von Deskriptoren in einem gruppierten, hierarchischen oder polyhierarchischen Thesaurus organisiert werden; hier ist auch eine effektive Terminologiekontrolle möglich. Der Zielsetzung dieser Arbeit entsprechend soll aber hier auf die sehr weitgehenden Möglichkeiten von LIDOS in diesem Bereich nicht weiter eingegangen werden.

8. Sonstige allgemeine Kriterien

8.1. LIDOS erwies sich bis auf die oben erwähnten Turbulenzen, die durch den Import größerer Datenmengen ausgelöst wurden, als durchweg stabil. Bedienungsfehler werden konsequent durch Fehlermeldungen aufgefangen.

8.2. Die Datensicherung ist unter LIDOS allein Sache des Anwenders. Das Handbuch verweist diesen auf das BACKUP-Kommando von MS-DOS, dessen Gebrauch durch die wenig durchsichtige Struktur der Dateien unter LIDOS nicht eben erleichtert wird.

8.3. Ein effektiver Passwortschutz ist unter LIDOS sehr einfach einzurichten; die Passwörter werden dabei nicht in einer über das Betriebssystem erreichbaren Liste verwaltet, sondern wirklich kodiert. Es sind drei Zugangsstufen realisierbar: Master-Zugriff, schreibend-/lesender Zugriff und nur-lesender Zugriff.

8.4. Im allgemeinen stellt das durchdachte und informative Menüsystem von LIDOS ein sehr effektives, allerdings bisweilen den Benutzer fast gängelndes Hilffsystem dar, das durch einige wenige informative Fehlermeldungen ergänzt wird. Weniger hilfreich sind die weiteren 190 nummerierten Fehlercodes, die jeweils einen Blick in die entsprechenden Handbucherläuterungen erforderlich machen. Besonders gelobt werden soll indessen das Handbuch: es läßt, was die Form der Präsentation, die Übersichtlichkeit, die schnelle Benutzbarkeit und meist auch die Qualität der Information angeht, wenig zu wünschen übrig.

8.5. LIDOS wird zu einem Grundpreis von 2 900 DM angeboten, bei Abnahme größerer Mengen sind Staffelpreise vorgesehen, es gibt jedoch keine Campuslizenzen. Realistischerweise ist dann noch die zusätzliche Anschaffung des Moduls LIDOS-download (490 DM) einzukalkulieren. Hochschuleinrichtungen erhalten einen Rabatt von 25% auf diese Preise.

E zusammenfassende Charakterisierung

LIDOS 3.0 ist ein äußerst bedienerfreundliches System für die selbständige Dokumentationsarbeit. Für den privaten Einzelanwender und für unabhängige Dokumentationsstellen, zumal wenn sie auf die Sacherschließung von Dokumenten Wert legen, ist es in seiner Kategorie wahrscheinlich das leistungsfähigste Produkt. LIDOS ist somit weit eher ein Informationsretrievalsystem denn ein Datenverwaltungssystem. Ein „Arbeitsmittel für den bibliothekarischen Alltag“ an der „kleineren bis mittelgroßen Spezialbibliothek“ oder „am Arbeitsplatz in großen Bibliotheken aller Typen“ jedoch ist LIDOS nicht. Gegen den dezentralen Einsatz in komplexen lokalen Systemen sprechen vor allem die enormen ergonomischen Unverträglichkeiten, die fehlenden bibliographischen Strukturierungsmöglichkeiten, die Probleme beim Datenaustausch und nicht zuletzt der Preis. Wie mit unter LIDOS erfaßten Daten in einem lokalen Datenpool effektiv gearbeitet werden könnte, ist schwer auszumalen.

Die eigentlichen Stärken von LIDOS, seine Möglichkeiten der Sacherschließung, die sehr guten Retrievalfunktionen und die integrierten Textverarbeitungsmöglichkeiten fallen hingegen bei einer Beurteilung im hier zur Diskussion stehenden Kontext wenig ins Gewicht.

3.3. TINman ■ TINlib

A Charakterisierung

Das Software-Paket TINman ist ein integriertes System für Erwerbung, Katalogisierung, Thesaurusverwaltung, Ausleihe, Zeitschriftenverwaltung und kann als microrechnergestützter OPAC (Online-Public-Access-Catalogue) eingesetzt werden. Aus Gründen der Vergleichbarkeit sollen in diesem Zusammenhang nur die Katalogisierungs-, Recherche- und (am Rande) die Thesaurusfunktion einbezogen werden.^{ss} Das gesamte Paket wird derzeit im Rahmen eines OBI-Projektes in allen Teilbereichen getestet und an bundesdeutsche Verhältnisse angepaßt, es wird sicher in naher Zukunft Gegenstand von Projektberichten sein.^{si5}

Kern des Systems TINman ist das Datenbanksystem TINMAN, das über ein hochentwickeltes Parametrierungs- und Programmierungssystem an die unterschiedlichsten Einsatzbedingungen anzupassen ist. So existiert auf der gleichen Basis ein weiteres Bibliothekssystem, PC/PALS.

Charakteristisch für TINman sind neben den integrierten Komponenten des Gesamtpaketes:

- eine sehr benutzerfreundliche und leistungsfähige Retrievalkomponente (vom angepeilten Einsatzgebiet OPAC her verständlich);
- weitgehend unzureichende oder zumindest nicht dokumentierte Druckfunktionen;
- weitgehende Möglichkeiten der Anpassung an verschiedene bibliographische Standards über ein allerdings nicht ganz einfach zu handhabendes Parametrierungssystem;

^{ss} Über Vor- und Nachteile integrierter Lösungen soll an dieser Stelle nicht nachgedacht werden. Soweit der potentielle Einsatz der hier untersuchten Softwarekategorie von solchen Überlegungen betroffen ist, in denen TINman und Allegro-C sicher zwei diametral unterschiedliche Konzeptionen repräsentieren, finden sich hierzu Bemerkungen unten in Kapitel 4.

⁸⁶ Projekttitle: „Einsatz eines Arbeitsplatzcomputers an Spezialbibliotheken mittlerer Oröönordnung“; vgl. dazu SCHMIDT. Die vorliegende Beschreibung entstammt dem Kontakt mit den Projektbeurteilern und -beratern.

– ein durchdachtes System hierarchisch differenzierter Zugriffsstufen, das in Funktion unterschiedlicher Anwendungs- und Arbeitserfordernisse konzipiert ist.

Das System besteht aus folgenden Komponenten:

- den eigentlichen Programmen (Kennung *.EXE, *.BAT und *.RUN);
- anwendungs- und schnittstellenorientierten Dateien als Befehls- und Hilfsdateien für die Kommunikation zwischen System und Rechner, für den Netzbetrieb, die Datenbankverwaltung etc. (ACT*.BIN, APT*.BIN, RCT*.BIN, DMT*.BIN, CIT*.BIN, PHANTOM*.BIN);
- der eigentlichen Datenbank in den Dateien FIXIT*.*.

B Entwicklung, Vertrieb

Das Programmpaket wurde von Peter Noerr und Mike White entwickelt, Hersteller ist die Firma IME,⁸⁷ für den Vertrieb in der Bundesrepublik ist die Firma UNISYS⁸⁸ zuständig.

Implementiert ist TINman derzeit in Deutschland nicht, das Programm wird aber von insgesamt ca. 70 Anwendern in Europa eingesetzt und ist auch in Netzwerken implementiert (so etwa in Florenz).

TINman ist in der englischen Originalversion hinsichtlich der Kernfunktionen fertig entwickelt, wird aber natürlich weiter gepflegt. Allerdings ist der tatsächliche Umfang der schon realisierten Funktionen mitunter sehr schwer einzuschätzen, da die Dokumentation viele Möglichkeiten des Systems einfach nicht erwähnt. Im Gespräch mit Herrn Habermann (OBI) stellte sich heraus, daß manch unverhoffte Funktion sich bei TINman nach entsprechend langem Probieren als realisiert erweist, so möglicherweise auch im Bereich der gegenwärtig als unzureichend empfundenen Druckfunktionen. Doch kann ein solches Herumprobieren natürlich dem Anwender nicht zugemutet werden: ausgegangen wurde daher bei der Beschreibung von den tatsächlich dokumentierten und klar zugänglichen Funktionen; daß dabei der

⁸⁷ IME Information Management/Engineering Ltd., 14-16 Farringdon Lane, GB London IDR 3AU, England.

⁸⁸ UNISYS Deutschland. Postfach 1110, 6231 Sulzbach.

eine oder andere Schatz ungehoben blieb, ist nicht auszuschließen. Getestet wurde TINman in zwei Versionen: an der Universitätsbibliothek Heidelberg konnte im März 1988 die noch weitgehend im Originalzustand befindliche englische Version erprobt werden, die neueste, an die Vorgaben des OBI-Projektes zumindest im Katalogisierungsbereich weitgehend angepaßte Version (3.0) wurde im Januar 1989 getestet.

C Literatur

Abgesehen von einem kurzen Abschnitt in dem Beitrag von SCHMIDT ist eine Kurzbeschreibung von TINman in der „EDV-Programm-Börse“ der Zeitschrift ABI-Technik⁸⁹ erschienen. Die Entwickler haben das System in den Aufsätzen BIVINS/NOERR und NOERR beschrieben.

Außerdem existiert ein ausführliches Handbuch in englischer Sprache.⁹⁰

D Beschreibung

1. Hardwareseitige und allgemein EDV-technische Bedingungen

1.1. TINman ist auf allen zum IBM-PC/XT und AT kompatiblen Rechnern und auf dem Siemens PC-D lauffähig. Ein AT-Rechner ist wegen des Laufzeitverhaltens dringend zu empfehlen.

1.2. TINman belegt ca. 250 KB Hauptspeicher. Erforderlich ist also ein Ausbau auf mindestens 512 KB. Die Program!1struktur von TINman läßt bei Einsatz einer RAM-Disk durchaus Laufzeitgewinnen erwarten.

1.3. TINman ist mehrplatzfähig (bis zu sieben Benutzer im Multitasking-Betrieb) und voll netzwerkfähig.

⁸⁹ ABI-Technik 7. 1987, 4; S. 379.

⁹⁰ TINman system documentation. Section A-G. London: IME 1985. Dem Verfasser lag eine im Jahr 1987 teilweise aktualisierte Fassung vor, die allerdings c,r,h,i,h,l,r,l,w Ulcken und Mängel aufwies.

1.4. TINman ist derzeit unter MS-DOS (ab Version 2.0) verfügbar, eine UNIX-Version befindet sich in der Entwicklung.

1.5. TINman ist in der eigens konzipierten Programmiersprache FIXITAL erstellt, was geringe Portabilität verspricht.

1.6. Die Architektur des Systems ist insgesamt übersichtlich, die interne Datenbankstruktur hingegen praktisch unzugänglich – letzteres ist von den Entwicklern durchaus beabsichtigt. Leider lagen dem Verfasser die Handbuchteile J (System programmers manual) und M (Beschreibung der FIXITAL-Sprache) nicht vor, die auch nicht Bestandteil der systembegleitenden Dokumentation sind. Eine Lieferung des Quellcodes wird ausdrücklich ausgeschlossen.

1.7. Eine online-Schnittstelle für den Datenaustausch existiert nicht, soll aber vorgesehen sein.

2. Datenbanktechnische Spezifikationen

2.1. Grenzwerte:

2.1.1. Die maximale Anzahl der Datensätze dürfte in der Praxis durch den zur Verfügung stehenden Massenspeicher begrenzt sein, die Beschreibung in ABI-Technik nennt „etwa 100000“ Datensätze als Höchstzahl.⁹¹ Eine Besonderheit von TINman ist, daß die Dateigröße vom Systempfleger bestimmt werden muß; wenn die Obergrenze der definierten Speicherkapazität erreicht ist, kann diese Kapazität zwar manuell erweitert werden, in der Praxis dürfte dies Verfahren aber nicht immer ganz ohne Probleme abgehen.

2.1.2. Die Anzahl der Zeichen pro Datensatz unterliegt keiner Beschränkung.

2.1.3. Die Feldzahl unterliegt keiner Beschränkung.

2.1.4. Ein Feld kann maximal 2010 Zeichen lang sein.

2.1.5. Es kann immer nur nach einem Feld (dem Indexfeld) gleichzeitig geordnet werden.

⁹¹ ABI-Technik 7. 1987; 4. S. 379.

2.2. TINman basiert auf einer eigenen, relational organisierten Mehrdateienstruktur unter Verwendung von getrennt aktualisierbaren Stammdateien (für Namensansetzungen u.ä.), der Direktzugriff geschieht über Indexfelder in den jeweiligen Datensätzen, die von einer Indexdatei (FIXIT*.SET) verwaltet werden. Neben den üblichen weiteren Datenfeldern erlauben Verbindungsfelder („link-fields“) in jedem Datensatz den Direktzugriff auf andere Datensätze über das durch den Feldtyp bestimmte Kriterium. Maximal 256 Dateien werden logisch zu Datenbanken (sets) zusammengefaßt, zusammengehörige Dateien werden durch eine dreistellige Identifikationsnummer im Dateinamen vor dem Trennpunkt kenntlich gemacht (so gehören alle Dateien FIXITool.* zusammen, wobei diese in der Namenskenung von 000 bis 255 durchnumeriert sind). Innerhalb einer Datenbankeinheit enthalten die Dateien FIXIT*.000 bis FIXIT*.015 Informationen zur Dateioorganisation und adressierung, benutzbar sind also immer nur die Dateinummern *.016 bis *.255, diese Dateien werden vom System bei Bedarf automatisch erzeugt und numeriert.

2.3. Feld- und Satzlängen sind im Rahmen der Grenzwerte flexibel.

2.4. Felder können frei eingerichtet und benannt werden, dies geschieht über „komplexe, allerdings nicht immer ganz einfache Parametrierungen in den 'Profile Formats'“.

2.5. Feldlängen sind nicht determinierbar.

2.6. Der Speicherbedarf konnte nicht ermittelt werden. Die Beschreibung in ABI-Technik gibt je 10000 Titel einen Massenspeicherbedarf von 10-20 Megabyte an.

3. Regelwerks- und Kategorienanforderungen

3.1. Eine Differenzierung und Strukturierung der Datensätze nach dem 'DBI-Schema' ist möglich.

3.2. Hierarchisch strukturierte Aufnahmen sind möglich.

3.3. Formalhierarchischen Verknüpfungen sind im vollen Umfang möglich, doch wird deren Wert durch die unten darzustellenden Unklarheiten in der Benutzerführung, den Schwächen bei der Verknüpfung von Titel- und Ansetzungsdateien und durch die allgemein sehr langen Antwortzeiten der Funktionen erheblich geschmälert.

4. Dateneingabe, Importfunktion

4.1. Ergonomie, Akzeptanz:

4.1.1. Erscheinungsbild und Anordnung der Eingabefelder im edit-Modus, einem bildschirmorientierten Editor, sind funktional und ansprechend, doch werden in der aktuellen Aufnahme nicht benötigte Felder leider nicht aus der Maske getilgt, die dadurch leicht unübersichtlich wird. Das System unterstützt einen Farbbildschirm und nutzt diesen intelligent aus, ohne in Spielereien zu verfallen: dies erwies sich beim Test als nicht unerhebliche Erleichterung für die Orientierung auf dem Bildschirm.

4.1.2. Die Editormaske kann hinsichtlich der graphischen Präsentation der Eingabefelder nicht umgestaltet werden, wohl aber können Zahl, Benennung und Reihenfolge der Felder bestimmt werden.

4.1.3. Es können feldbezogenen Daten aus demselben oder aus anderen Sätzen kopiert werden, ganze Sätze können nicht kopiert werden.

4.1.4. An sonstigen Erfassungshilfen existiert noch die sogenannte „Validierfunktion“, die über Fenstertechnik das Einblenden anderer Datensätze, von authority-files zur Ansetzungskontrolle und generell alle Suchvorgänge in der aktuellen Datenbank ermöglicht.⁹² Aus den im Fenster eingeblendeten Datensätzen und Ansetzungsdateien können auf Tastendruck markierte Daten übernommen werden.

Außerdem ist positiv hervorzuheben, daß die Editorfunktionen weitgehend über konsequente Belegung der Funktions- und Sondertasten der PC-Tastatur gesteuert werden, was die Einarbeitung erleichtert.

Negativ hingegen muß folgendes zur Benutzerführung angemerkt werden: TINman erlaubt den Zugriff auf und das Redigieren von Daten mit sehr unterschiedlichem Bearbeiterstatus, auch können Eingabedateien (aufgrund der 'navigating'-Funktion, s.u.) von verschiedenen Menupositionen her erreicht werden, wobei dann auch der Zugriffstatus und die Ediermöglichkeiten sehr unterschiedlich sind. Leider unterstützt das System aber den Anwender minimal bei der Nutzung dieser an sich flexiblen und differenzierten Zugriffsmöglichkeiten: in den

seltensten Fällen ist erkennbar, zu welchem Eingabebereich eine Datenmaske tatsächlich gehört, welchen Zugriffs- und Bearbeitungsstatus der Anwender zu diesem Zeitpunkt hat; dieser grundsätzliche Mangel erzwingt (unter Umständen zeitraubendes) Probieren, bis der Anwender dann schließlich durch den erfolgreichen Abschluß etwa eines Speichervorgangs oder aber durch eine Fehlermeldung des Systems aus seiner Unsicherheit erlöst wird.

Hinzu kommen die unten (4.5.) angesprochenen langen Antwortzeiten des Systems: sie potenzieren in vielen Fällen diese Schwächen in der Benutzerführung bis zur Frustration.

4.2. Routinen zur Sicherung der Datenhomogenität:

4.2.1. Die oben angesprochene Validierungsfunktion stellt einen Ansatz zur authority-control dar, für die ja auch mit der Mehrdateienstruktur von TINman die elementare Voraussetzung gegeben ist. Prinzipiell können Daten aus Ansetzungsdateien übernommen werden, wenn entsprechende 'validation profiles' erstellt sind, können die Inhalte bestimmter Felder automatisch auf Ansetzungskonformität überprüft werden. Diese Funktion war in der zuerst im März 1988 an der Universitätsbibliothek Heidelberg getesteten Version (im wesentlichen noch die englische Originalversion) konsistent; in der neuesten vorliegenden, im Rahmen des OBI-Projektes erarbeiteten Anpassung des Systems ist diese Direktverknüpfung von Titel- und Ansetzungsdateien aus nicht ganz klaren Gründen zugunsten einer indirekten Verknüpfung über eine Zwischendatei aufgegeben worden; diese Zwischendatei nimmt Verknüpfungsinformationen aus den Titel- und Stammdateien auf und stellt so eine Verbindung zwischen beiden Datenbereichen her; dies ist aber doch recht anfällig! So beeinflusst etwa das Löschen eines Autors aus der Namensdatei keineswegs die Titeldatei: dort sind die mit dem Autor 'verknüpften' Titel mitsamt dessen Namen unverändert vorhanden; umgekehrt analog

verhält es sich beim Löschen eines Titels aus der Titeldatei: der Autor wird in der Namensdatei weiter als der Autor des in der Titeldatei gelöschten Werkes geführt. Das Edieren der Verknüpfungsdatei ist möglich (warum?), doch sollte von dieser Möglichkeit kein Gebrauch gemacht werden: der völlige Verlust der Verknüpfungsinformationen ist die Folge. Die Liste solcher Ungereimtheiten ließe sich verlängern; sie haben ihre gemeinsame Ursache darin, daß die 'Verknüpfung' im gegenwärtigen Zustand des Systems einfach

durch die Kopie der verknüpfungsrelevanten Informationen in die Verkettungsdatei realisiert ist.

TINman wartet mit einer Doublettenkontrollfunktion auf, die jedoch in der gegenwärtigen Form keine große Erleichterung darstellt: die Kontrolle auf doublette Eingabe erfolgt ausschließlich über die Titeltategorie als 'key field',⁹³ was dazu führt, daß zum einen Aufnahmen mit gleicher Titelformulierung (aber etwa anderen Ausgabevermerken o. ä.) vom System nicht angenommen werden, und daß andererseits die erneute Aufnahme desselben Titels schon dann möglich ist, wenn der Titel mit an sich unerheblichen Schreibfehlern eingegeben wird.

4.2.2. Plausibilitätskontrollen sind global und feldbezogen über Änderungen in den 'storage profiles' völlig frei realisierbar. Standardmäßig existiert an feldbezogenen Kontrollen nur die Berechnung der ISBN-Prüfziffer.

4.2.3. Pflichtkategorien und fakultative Eingabefelder werden unterschieden. Der Status der Felder ist vom Anwender über Parametrierungen in den 'profile formats' frei bestimmbar.

4.2.4. Regelwerkshilfen sind derzeit nicht realisiert. Sie sind aber leicht über frei erstellbare Hilfstexte integrierbar und wären dann im 'Validiermodus' als Bildschirmfenster situationsbezogen verfügbar.⁹⁴

4.3. Datenfelder sind beliebig oft belegbar und können ad hoc dupliziert werden, einmal (auch versehentlich) duplizierte Felder können allerdings nicht mehr aus der Eingabemaske entfernt werden, was diese nicht eben übersichtlicher macht.

4.4. Es ist der Zeichensatz des jeweiligen Rechners voll verfügbar.

4.5. Das Laufzeitverhalten konnte nicht unter realistischen Bedingungen, d. h. mit großen Datenmengen, getestet werden. Doch verspricht schon das Antwortzeitverhalten bei zweistelligen Datensatzmengen für den Ernstfall nichts Gutes: schon bei dieser geringen Datensatzmenge

⁹³ Hier handelt es sich auch nach der Einschätzung des OBI-Projektes um einen systemimmanenten Fehler, der nicht durch Parametrisierungsarbeiten, sondern allenfalls durch die Firm a IME zu beheben ist.

⁹⁴ Entsprechende Versuche werden im Rahmen des OBI-Projektes gemacht und sind bislang in der Universitätsbibliothek Heidelberg vielversprechend verlaufen.

benötigt TINman zum Aufbau von Suchlisten, in vielen Fällen sogar zum Auffinden von Zielinformationen, teilweise weit über 10 Sekunden; bei häufigem Funktions und Datenbereichswechsel wird also die Geduld und Arbeitszeit des Anwenders arg strapaziert.

4.6. Importfunktionen:

4.6.1. Leider erwies sich beim Testen der Originalversion im März 1988 gerade die Importfunktion, die ja auch als Grundlage für die im Test vorgesehenen Laufzeituntersuchungen eine Schlüsselstellung einnimmt, als defekt.⁹⁵ Beim Test in Heidelberg ließ sich die Funktion zwar aktivieren, war aber aus weder vom Verfasser noch vom Leiter des OBI-Projektes, Ronald M. Schmidt, zu eruierten Gründen außerstande, die völlig korrekten MARC-Feldkennungen zu erkennen. Langwierige Versuche, den Fehler über Parametrierungsänderungen zu beheben, brachten kein Ergebnis. Die Erarbeitung einer auf die im Rahmen des OBI-Projektes erstellten Anpassungen bezogenen Importparametrisierung ist in Arbeit.

Die Funktion kann daher an dieser Stelle nur theoretisch beschrieben werden. Im Prinzip ist sie komfortabel und bedienerfreundlich: sie erfolgt komplett menugesteuert, das Format der zu importierenden Daten wird in einer Liste mit dem Cursor markiert und durch Betätigen der Return-Taste ausgewählt, einzig der Name der zu importierenden Datei und gegebenenfalls die Anzahl der zu importierenden Sätze müssen eingegeben werden. Der Import kann dann entweder im 'Vordergrund' (mit Bildschirmanzeige aber ohne Eingreifen des Anwenders), oder interaktiv (hier entscheidet der Anwender im Einzelfall darüber, ob ein Datensatz eingespielt werden soll oder nicht) oder im 'Hintergrund' geschehen (dann kann während des Importvorgangs mit beliebigen anderen TINman-Funktionen gearbeitet werden. Eine Ausgabe der Meldungen über gefundene Fehler in den zu importierenden Daten auf Drucker oder Bildschirm ist möglich, zusätzlich kann das System statistische Angaben über den Verlauf des Vorgangs (Anzahl der Datensätze, Dauer des Imports etc.) ausgeben. Die zu importierenden Daten müssen im ASCII-Format vorliegen (auch Hex-ASCII wird akzeptiert), der Import von Binärdateien soll in Zukunft möglich sein.

⁹⁵ Es hätte dann also zumindest in der erwähnten 'Programmbörse' von ABI-Techne nicht angegeben werden dürfen, der dort beschriebene und für die Konzeption nur Tustlufc im Rahmen dieser Arbeit eben zu rundgcglnte Punktica INIIIIIIIIH Noi „xdt Mitte 1987" vcrllgblr.

Die Erstellung der für den Import notwendigen Importtabellen hingegen ist eine mühselige, wenn auch vom System so gut als möglich erleichterte Arbeit. Es sind drei Tabellen zu definieren (master control table, input details table und conversion table), die Informationen über die Struktur der Ausgangsdatei, Feldzuweisungen und gegebenenfalls auch die Festlegung von Austauschvorgängen enthalten. Die hier notwendigen Angaben sind zusammen mit der Befehlssyntax im 'Systems Designer's Manual' verständlich, jedoch sehr knapp und leider ohne Beispiele erläutert.

4.6.2. Die Importfunktion von TINman ist ausreichend flexibel und äußerst komfortabel. Die Erstellung von Umsetzungstabellen ist wie bei allen vergleichbaren Systemen nicht ganz einfach, sie wird aber unter TINman zumindest nicht durch unnötige zusätzliche Komplikationen erschwert.

4.6.3. Grundvoraussetzungen für den online-Import sind durch die Importschnittstelle von TINman gegeben.

4.6.4. Laufzeiten konnten nicht ermittelt werden.

5. Datenausgabe, Exportmodus

Die in diesem Bereich aufzuführenden Schwächen von TINman sind möglicherweise in der Dokumentation des Systems begründet. Sollten aber in verborgenen Winkeln noch ungeahnte Möglichkeiten von TINman schlummern, so sind diese jedenfalls derzeit nicht nutzbar, weil unzugänglich.

Diese Mängel sind so weitreichend, daß auch im Rahmen des DBI-Projektes derzeit erwogen wird, die eigentliche Datenausgabe nicht über die TINman-internen Funktionen, sondern über ein neu zu programmierendes Zusatzmodul zu realisieren.

5.1. Zetteldruck, Bildschirmausgabe einzelner Aufnahmen:

5.1.1. Die Gestaltung der Ausgabe ist über die Freiheit bei der Felddefinition und -benennung sowie die Möglichkeit, die

Reihenfolge der

beeinflussen! Der Zetteldruck nach ISBD etwa ist also mit TINman nicht möglich.

5.1.3. Entfällt.

5.1.4. Nebeneintragungen können nicht erzeugt werden.

5.1.5. Entfällt.

5.1.6. Die Druckfunktionen von TINman sind insgesamt völlig unzureichend. Es können nur der Bildschirminhalt (über die wohl deshalb auch vom Handbuch genannte prtsr-Funktion des PC), einzelne Datensätze im normalen Anzeige/Edierformat oder mehrere Datensätze in Form einer schlichten Liste ausgegeben werden. In diesem beschränkten Rahmen ist die Funktion dann allerdings komfortabel und wird über Funktionstastenbelegung gesteuert.

5.2. Listendruck, Exportmodus:

5.2.1. Die Ordnungsfunktionen bei gereihter Datenausgabe sind unzureichend. Auf komplexe Ordnungsfunktionen ist hier nicht einzugehen, da es schon schon bei ganz elementaren Ordnungsvoraussetzungen Mängel gibt: TINman sortiert rein zeichenkettenorientiert, so daß etwa eine Namensform mit und ohne abschließendem Leerzeichen am Ende einen je unterschiedlichen Ordnungswert hat. Zudem sortiert TINman hierbei einfach nach der Reihenfolge der Zeichen in der ASCII-Tabelle: ein kleines „a“ hat einen höheren Ordnungswert als ein großes „Z“, deutsche Umlaute stehen nach allen anderen Zeichen des lateinischen Alphabets, gleiches gilt für andere nationale Sonderzeichen etc. Möglicherweise sind diese Mängel noch durch Anpassungen in den Parametrisierungstabellen aufzufangen – in der gegenwärtigen Form jedenfalls sind damit schon einige der Voraussetzungen für komplexere Ordnungsvorgänge nicht gegeben.

5.2.2. Die Gestaltung der Listenausgabe ist menugesteuert, sie bietet minimale Möglichkeiten. Eingestellt werden können Zeilenlänge, Seitenwechsel nach jedem Datensatz (Ja/Nein) und der Abstand zwischen den Datensätzen.

ies g<iNc lihih lUbr Parametrierungen in den 'retrieval profiles' und den 'storage profiles'.

5.2.3. Der Exportmodus von TINman ist zugleich denkbar einfach und wenig leistungsfähig : nach Auswählen der zu exportierenden TINman-Daten wird durch Betätigung einer Funktionstaste der Exportmodus aktiviert, es wird der Name der MS-DOS Zielfeld angegeben (falls diese schon existiert, kann sie überschrieben oder ergänzt werden), anschließend werden die Daten exportiert.

5.2.4. Die Zeiten für die Funktionshandhabung sind, von den allgemein langsamen Reaktionszeiten des Systems einmal abgesehen, minimal : aufgrund der Aufrufmöglichkeit über Funktionstasten liegen sie für den eigentlichen Export-/Druckvorgang unter 10 Sekunden – allerdings muß diesem bei der Ausgabe mehrerer Datensätze die Bildung einer entsprechenden Suchmenge vorangehen.

5.2.5. Das Exportprodukt von TINman ist eine unformatierte ASCII-Datei, die problemlos mit jeder Textverarbeitungssoftware weiterverarbeitet werden kann.

5.2.6. Funktionen und Oberfläche sind im Rahmen des minimalen Leistungsumfanges maximal komfortabel.

5.2.7. Die Voraussetzungen für den online-Export scheinen gering zu sein.

6. Zugriffs- und Recherchfunktionen

Die ansonsten übliche Unterscheidung zwischen einem Schnellzugriffs- und einem Volltextsuchmodus hat im Falle von TINman nur sehr bedingt einen Wert, sie wird hier dennoch beibehalten, um die Übersichtlichkeit der Darstellung zu gewährleisten, vor allem aber ermöglicht sie den Vergleich mit anderen Systemen.

6.1. Schnellzugriffsmodus:

6.1.1. Die überaus leistungsfähige und komfortable Realisierung des Schnellzugriffs unter TINman ist eine der Stärken des Systems. Die Suche nach und der Zugriff auf Informationen sind dabei von zwei Grundtechniken bestimmt: dem 'browsing' und dem 'navigating'. Ersteres ist aus anderen Zusammenhängen bekannt, auch bei TINman ist unter 'browsing' das (cursorgesteuerte) Durchsuchen von Bildschirmseiten und die Auswahl von Dokumenten aus diesen Listen zu verstehen.

Ein solches 'browsing' ist anhand folgender Kategorien in alphanumerisch geordneten Listen, sogenannten 'current record sets', möglich:

- Autor,
- Klassifikationscode,
- ISBN/ISSN,
- Stichwort,
- Verlag,
- Zeitschriftentitel,
- Monographientitel.

Ist ein Dokument ausgewählt, so kann die Anzeige der zu dem Datensatz gehörigen Detailinformationen erfolgen. Will nun der Anwender anhand eines Feldinhaltes in einer solchen Titelaufnahme andere Aufnahmen mit gleichen Feldinhalten suchen, kommt die zweite Abfragetechnik, das 'navigating' zum Einsatz. Es wird hierbei mit dem Cursor ein Selektionspfeil vor das gewünschte Feld geführt. Nach Drücken der Return-Taste erscheint eine Liste aller über das selektierte Feld mit dem aktuellen Datensatz verbundenen Sätze auf dem Bildschirm.

Beim Suchen in langen 'current record sets' ist ein gezieltes Anwählen durch Eingabe von Suchwörtern mit beliebiger Maskierung möglich (sogenannte 'Skip-Suche'), bei der Suche in Zahlenlisten ist eine numerische Bereichseinschränkung möglich.

Zusätzlich existiert die Möglichkeit der „Abfragetechnik“. Dabei werden in Suchschablonen, die den Suchprofilen der aktuellen Datenbank entsprechen, feldweise Zeichenketten zur Suche eingegeben, die Verwendung logischer Operatoren ist dabei möglich.

6.1.2. Die Sekundärdatei FIXIT*.SET wird bei Neueingaben oder Änderungen automatisch aktualisiert.

6.1.3. Maskierungsmöglichkeiten sind durch die 'Skip-Suche' gegeben.

6.1.4. 'browsing' ist eine der beiden Grundtechniken des Systems.

6.1.5. Die jeweils letzte Suchmenge kann gespeichert, mit den Ergebnissen weiterer Recherchen kombiniert oder weiter eingeschränkt werden.

6.1.6. Laufzeiten konnten nicht unter realistischen Bedingungen ermittelt werden.

6.2. Volltextsuche:

Die Volltextsuche wird unter TINman als „Filtertechnik“ bezeichnet und funktioniert gleichsam ex negativo: sie eliminiert aus einem 'set' logisch alle Datensätze, die einen Suchausdruck nicht enthalten. Der Suchmodus ist unterschiedlich, je nachdem ob wort- oder zeichenkettenorientiert gesucht wird.

6.2.1. Die eigentliche Volltextsuche ist nicht über Formalkategorien eingrenzbar. Einen Ersatz stellt jedoch die „Abfragetechnik“ dar.⁹⁷

6.2.2. Bei der Wortsuche ist nur die Rechtsmaskierung möglich, gesucht werden können also nur Wörter und Wortanfänge; bei der Zeichenkettensuche ist die (hier einzig vorstellbare) Mittelmaskierung nicht möglich.

6.2.3. Zur Bildung komplexer Suchbegriffe können die logischen Operatoren AND, OR und NOT verwendet werden. Bei der Zusammensetzung der Ausdrücke können Klammern eingesetzt und mehrfach geschachtelt werden.

6.2.4. Laufzeiten konnten nicht ermittelt werden. Das Handbuch empfiehlt jedoch (wohl aus gegebenem Anlaß), Wort- und Stringsuchläufe mit der Filtertechnik nur bei kleinen Datenbanken oder im Notfall durchzuführen.⁹⁸

6.3. Komfort und Umfang der Funktionen sind vorbildlich, auch die Benutzeroberfläche ist ansprechend. Auch soll hier nochmals auf die Möglichkeit hingewiesen werden, die Ergebnisse maximal eines Suchlaufes zu speichern und dann mit den Ergebnissen eines weiteren Suchvorganges zu kombinieren, dabei sind die Suchtechniken frei kombinierbar.

⁹⁷ S. o. unter 6.1.1.

⁹⁸ Wie Anm. 90, S. 26.

7. Möglichkeiten der Sacherschließung

Es ist eine kontrollierte Sacherschließung unter Verwendung des vom System geführten Thesaurus möglich. Zusätzlich ist natürlich die Definition beliebiger Sacherschließungsfelder möglich, auf die dann auch wie oben beschrieben zugegriffen werden kann.

8. Sonstige allgemeine Kriterien

8.1. TINman erwies sich in allen funktionierenden Bereichen als weitgehend stabil. Durch Bedienungsfehler dürfte TINman nach Einschätzung des Verfassers kaum zum 'Absturz' zu bringen sein.

8.2. Die Datensicherung ist allein Sache des Anwenders. Aufgrund der unklaren Struktur der TINman-Dateien kommt hier nur eine Sicherung sämtlicher FIXIT-Dateien im backup-Verfahren oder auf streamer-tape in Frage, die sich in der Praxis als zeitaufwendig erweisen dürfte und zudem einen nicht unerheblichen Massenspeicherbedarf mit sich bringt.

8.3. Das System unterschiedlicher, völlig frei definierbarer Benutzerklassen stellt eine der Stärken von TINman dar und geht weit über einen hierarchischen Zugriffsschutz hinaus. Es können Benutzergruppen definiert werden, denen Zugang zu frei eingrenzbaeren Funktionsbereichen von TINman gewährt werden soll; dies geschieht durch Vergabe je eines Passwortes für jede Benutzergruppe.

8.4. Das System verfügt über frei anpaßbare (und in der Originalversion auch korrekte) Hilfsfunktionen,⁹⁹ wozu auch die konsequente, mnemotechnisch günstige Steuerung der Anwendungen durch Funktions- bzw. Sondertasten zu zählen ist. Allerdings wird auch deren Wert durch die für TINman charakteristischen langen Antwortzeiten geschmälert. Der Grad der Systemhilfen ist zudem über eine Einstellung der Hilfsniveaus in zehn Stufen bestimmbar. Allgemein wirken sich jedoch die oben angeführten Schwächen in der Benutzerführung sehr negativ aus.

⁹⁹ Diese sind bei der Anpassung durch das OBI-Projekt zum gegenwärtigen Zeitpunkt wohl noch vernachlässigt und zum Teil auch nicht korrekt.

Die (bis auf das Benutzerhandbuch) bislang nur in englischer Sprache vorliegende Begleitdokumentation ist in hierarchisch gegliederte Informationsbereiche aufgeteilt: das Benutzerhandbuch gibt rein funktionsorientierte Informationen zu Retrieval, Anzeige- und Edierfunktionen; das 'System Administrator's Manual' beschreibt die Erstellung, Erweiterung und Organisation von Datenbanken sowie Möglichkeiten der Benutzerschulung; das 'System Designer's Manual' hat vor allem das Parametrierungssystem zum Thema. Diese im Prinzip durchdachte Verteilung von Informationen erweist sich in der Praxis mitunter als hinderlich; oft ist nicht klar, an welcher Stelle welche Informationen zu suchen sind; diese Unklarheiten in der Dokumentation stellen ein Pendant zu den Schwächen in der Benutzerführung dar. Die Präsentation der Informationen ist im Benutzerhandbuch gut und beispielreich, in den anderen Teilen eher karg (aber korrekt), mit Beispielen wird hier gezeigt (wobei die Verfasser offensichtlich ein höheres Abstraktionsniveau der 'Administratoren' bzw. 'Designer' voraussetzen). Nach den Testerfahrungen ist aber davon auszugehen, daß das Handbuch in seiner vorliegenden Form ganz erhebliche Lücken aufweist und manche Möglichkeiten von TINman gar nicht beschreibt.

8.5. TINman wird in der angepaßten Version nach Abschluß des DBI-Projektes der momentanen Planung nach für eine Zeit von einem bis zwei Jahren an Spezialbibliotheken zu einem Preis von 5 000-6 000 DM abgegeben werden, die Abgabekonditionen für universitäre Interessen stehen noch nicht fest, an Sonderbedingungen (und möglicherweise auch Campuslizenzen) für diese Zielgruppe ist aber gedacht.¹⁰⁰

E zusammenfassende Charakterisierung

TINman/TINlib ist ein weitgehend benutzerfreundliches und in der Gestaltung ansprechendes Paket zur microrechnergestützten Integration von bibliothekarischen Tätigkeiten. Seine 'Stärken' und 'Schwächen' stehen wohl mit den zumindest bei den Handbuchverfassern im Vordergrund befindlichen OPAC-Funktionen von TINman in Zusammenhang: Die Retrievalkomponente ist hochentwickelt, sehr differenziert

und zugleich benutzerfreundlich – hingegen wurde etwa die Druckausgabe in einer Weise vernachlässigt, die den Einsatz von TINman an einer Bibliothek mit weiterzuführendem konventionellen Katalog nicht zuläßt.¹⁰¹

TINman ist angesichts seines Leistungsumfanges als integriertes System preiswert.

Beim Einsatz von TINman entsteht eine sehr enge Bindung an die Firma IME, die trotz der sehr weitgehenden Parametrierungsmöglichkeiten schnell zu einer gefährlichen Abhängigkeit werden könnte. Später einmal gewünschte Zusatzfunktionen, Funktionsänderungen oder auch die programmtechnische Einbindung von TINman in komplexe lokale Lösungen: in vielen dieser Fälle wird man auf die Firma und deren Programmierkapazitäten angewiesen sein. Nicht umsonst ist ja der Quellcode der Programme kategorisch von der Lieferung ausgeschlossen, auch der Einblick in die Datenbankstrukturen ist nicht möglich. Man wende hier nicht ein, TINman erfülle in seiner jetzigen Form alle derzeit nur denkbaren Anforderungen (was zudem nicht zutrifft), Erweiterungen und Modifikationen seien also kein Thema: daß gerade bei EDV-Lösungen mit deren enormer innovatorischer Dynamik im Anwendungskontext überhaupt erst viele weitere Wünsche wach werden, die vorher gar nicht vorstellbar waren, daß hier salopp gesprochen oft der Appetit beim Essen kommt, müßten die Erfahrungen der Bibliotheksautomatisierung in den siebziger Jahren gezeigt haben. Eben diese Erfahrungen aus dem Großrechnerbereich müssen aber auch in anderer Hinsicht nachdenklich stimmen: daß die dort immer noch bestehende Firmenbindungen (vor allem über die Hardware) den deutschen Hochschulbibliotheken im Laufe der Zeit durchaus auch Unannehmlichkeiten beschert hat, muß hier nicht weiter ausgeführt werden.

¹⁰⁰ Mündliche Information von Ronald M. Schmidt (in Ergänzung der Informationen in HABERMANN, S. 841).

¹⁰¹ Doch 1,111 111(!) die oben gemachte Einschränkung hinsichtlich des DokumentationsNnt 11nd1111 vni 1 'IN11rnnl